ARMY RESEARCH LABORATORY

A Hardware Platform for Characterizing and Validating 1-Dimensional Optical Systems

by Thomas Kottke

ARL-TR-7089 **September 2014**

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory
Aberdeen Proving Ground, MD 21005

ARL-TR-7089 **September 2014**

A Hardware Platform for Characterizing and Validating 1-Dimensional Optical Systems

Thomas Kottke
Weapons and Materials Research Directorate, ARL

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
September 2014	Final	1 November 2010–31 May 2014

4. TITLE AND SUBTITLE
A Hardware Platform for Characterizing and Validating 1-Dimensional Optical Systems

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Thomas Kottke

5d. PROJECT NUMBER
AH60

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
US Army Research Laboratory
ATTN: RDRL-WMP-A
Aberdeen Proving Ground, MD 21005-5069

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-7089

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This report presents an optical hardware platform that is applicable to sensor systems utilizing a 1-dimensional (1-D) linear array. A basic goal in creating this apparatus is to provide a system that can simultaneously acquire, record, and analyze optical linear array data. The details of, and interactions between, the major hardware components, including the optical linear array, memory device, and microcontroller, are presented in detail along with ancillary subsystems that support and enhance the functionality of this apparatus. Specifics concerning the design and fabrication of the hardware prototype are highlighted. The operation of the optical linear array hardware platform is explained in detail, including data acquisition, reformatting, analysis, monitoring, display, short-term onboard storage, and off-platform downloading. Finally, a simple demonstration of the apparatus' functionality is presented.

15. SUBJECT TERMS
optical, linear array, data logger, data recorder optical, linear array, data logger, data recorder optical, linear array, data logger, data recorder

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Thomas Kottke
Unclassified	Unclassified	Unclassified	UU	48	19b. TELEPHONE NUMBER (*Include area code*) 410-278-2557

Contents

List of Figures

Acknowledgments

The author is delighted to thank Dr Müge Fermen-Coker of the Lethality Mechanisms Branch (LMB) of the Lethality Division (LD) for financial assistance with this project. In addition, the author would like to thank Dr George M Thomson of the Applied Physics Branch (APB) of the Protection Division (PD) for many helpful discussions, for guidance, for technically reviewing this manuscript, and for general good cheer. Finally, the author would like to thank the editors at SpecPro Technical Services for editorially reviewing this report.

1. Introduction

The Applied Physics Branch (APB) of the Weapons and Materials Research Directorate (WMRD) teamed with the Lethal Mechanisms Branch (LMB) to investigate novel adaptive fuze concepts for multiuse, flexible munitions. The fuze concept investigated utilizes a linear optical array sensor to view the topography of the approaching impact zone. For proof-of-principle laboratory experiments, a bread-board sensor and data collection system was created to gather fuze data to postprocess after the event. This non-real-time setup was sufficient to demonstrate the fuze's potential and to generate interest in additional real-time experiments. To characterize the performance of the fuze, it was desirable to record all the available data from the linear optical array for later analysis to develop optimal data processing algorithms. However, to validate the performance of the fuze, it was necessary to analyze the acquired data in real time to obtain a triggering solution. The requirements of simultaneous data logging and data processing can be burdensome—particularly at high-data acquisition rates.

This report presents the resulting hardware platform that enables simultaneous data acquisition, logging, and processing. Although created for the adaptive fuze study, this research apparatus has general applicability to the development of sensors that utilize 1-dimensional (1-D) optical arrays. First, the major optical and electronic hardware components that were chosen for this platform are presented. Next the integration of these major components into an efficient, multifunctional system is discussed highlighting architectural choices and modes of operation. Then, the ancillary, supporting electronic components that facilitate the operation of the platform is described. Finally, the operation and performance of the system is presented and demonstrated. Relevant electronic circuit diagrams, hardware details, and fabrication techniques are also included.

2. Major Hardware Components

2.1 Optical Linear Array

The choice of the particular optical linear array to use in this platform was based primarily on the desire to create an adaptive fuze that was both rugged and economical. Both features were satisfied by the TAOS TSL3301-LF linear optical sensor array. This electro-optic device combines 102 black-and-white light-sensing elements with 8-bit analog-to-digital conversion and serial communication circuitry in a convenient, 8-pin package.[1] Figure 1 illustrates this device and presents the physical layout, pin out, and designation of the light-sensing pixels. Each pixel is 85 μm long in the dimension transverse to the array axis and 77 μm long in the dimension parallel to the array axis. Furthermore, each pixel center is 85 μm from the centers of the

adjoining pixels. This geometry yields a pixel density of 120 dots per cm (300 dots per inch) and a total linear array dimension of approximately 0.9 cm. Figure 1 also highlights the fact that the 102 pixels are subdivided into 3 groups of 34, with each subgroup having independent offset and gain settings.

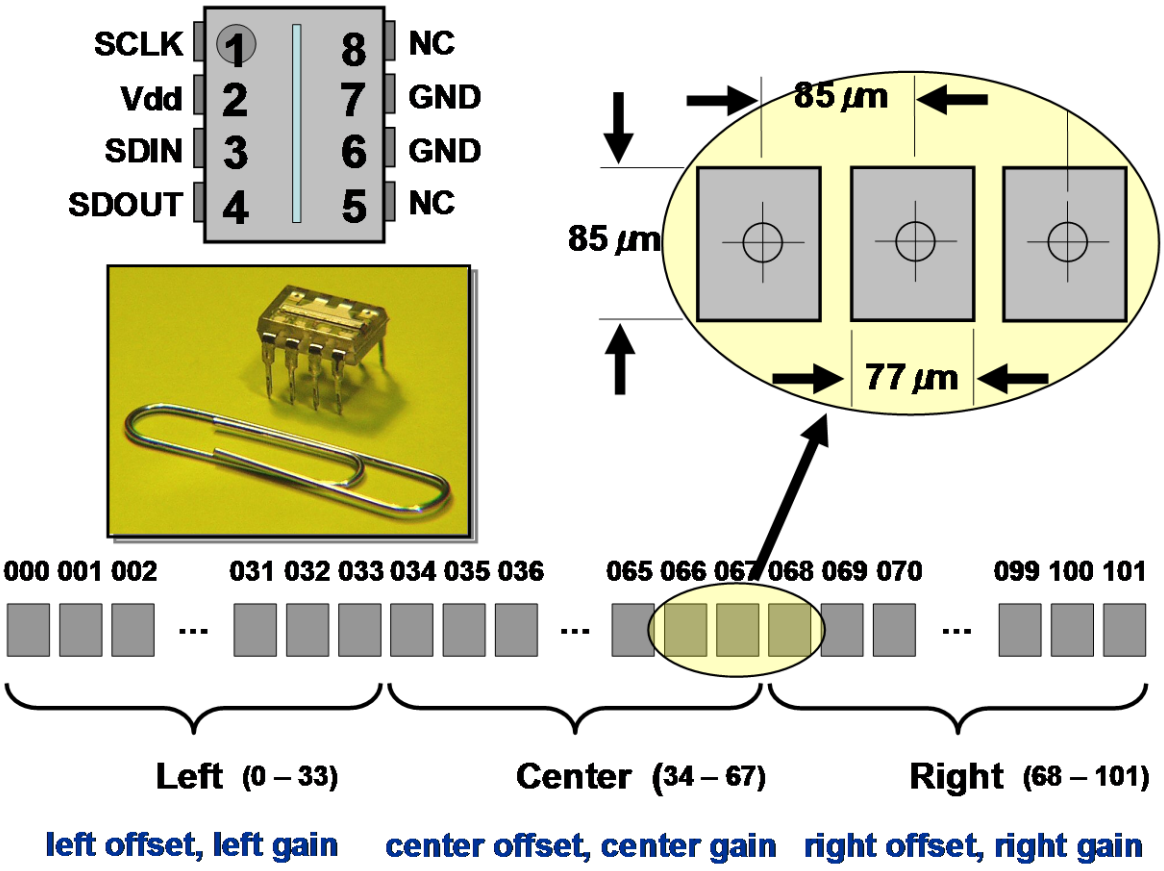

Fig. 1 TSL3301-LF linear optical array package and pixel geometry

The serial communications interface of the TSL3301-LF yields a device with an extremely low pin count. In addition to the 3 power-supply pins, there are only 3 other functional pins: a serial data in (SDIN) pin; a serial data out (SDOUT) pin; and a serial clock (SCLK) pin. Serial communications follow a universal synchronous and asynchronous receiver transmitter (USART) type format[2] with one start bit, 8 data bits, and 1 stop bit. As illustrated in Fig. 2, data are clocked into the linear array on the rising edge of the serial clock pulse SCLK and out on the falling edge of SCLK beginning with the start bit, followed by the 8 data bits from the least significant to the most significant, and ending with the stop bit. Conveniently, the receive and transmit functions of the TSL3301-LF are independent, which allows simultaneous communications to and from the linear array. This feature will prove useful in allowing the platform to simultaneously serve as a data logger and as data processor.

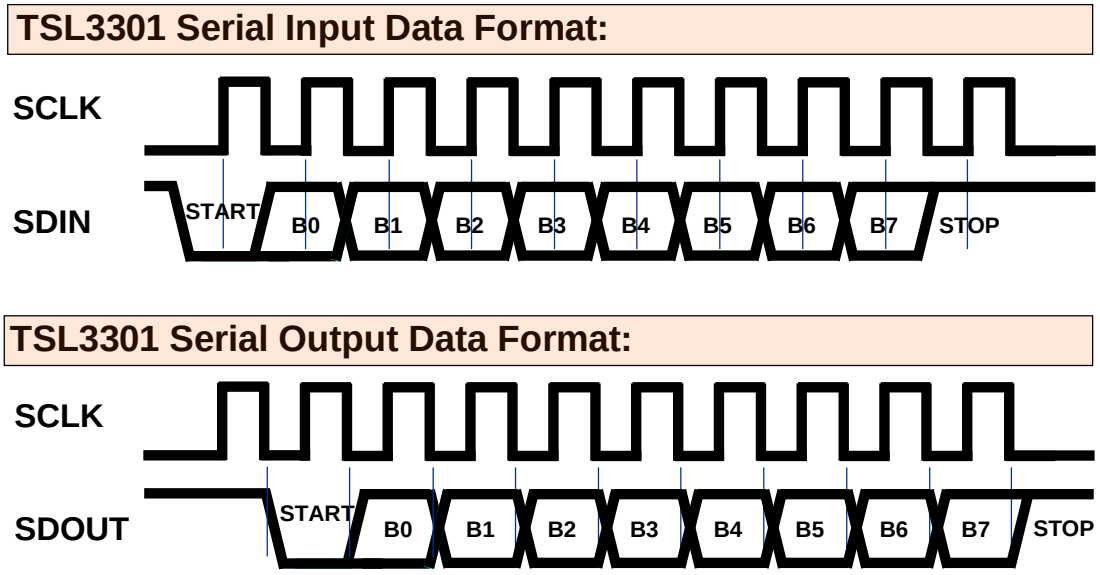

Fig. 2 TSL3301-LF linear optical array serial data formats

The TSL3301-LF linear optical array is a servant device, which means it must rely on some master device to provide the necessary clock and data input signals. Valid data input to this device includes 10 commands that control device initiation, device configuration, data acquisition, and data output. Valid data output from this device includes requested configuration parameters and measured pixel light levels. The maximum SCLK frequency is 10 MHz. With each pixel value output requiring 10 clock cycles (1 start + 8 data + 1 stop) the maximum pixel data output rate is 1 MHz. With a total of 102 pixel values output per sample, the TSL3301-LF sampling rate is slightly less than 10 kHz with 128 bytes of data transferred for each linear array sample. Clock cycles dedicated to device control and overhead will further decrease this sampling rate. Of course the goal of this effort is to obtain the highest possible sampling rate.

2.2 Memory

Some form of memory is obviously required if the hardware platform is to serve as a data logger for the linear optic array. A variety of memory types are available. Static random access memory (SRAM) offers the advantages of a near infinite number of available read/write cycles and high-speed operation. However, despite the word “static” in its name, SRAM devices are in fact volatile. The retention of stored data in SRAM devices relies on the availability of an uninterrupted supply power. (The term, static, merely differentiates this bistable memory category from dynamic random access memory [RAM], which must be periodically refreshed to retain data.) A momentary loss of power will clear the data recorded in an SRAM memory device. The loss of data resulting from a loss of power can be problematic for ballistic data recorders that are routinely subjected to extreme environments. Nonvolatile memory devices are available, but they generally support a limited number of read/write cycles and, thus, are subject to simply wearing out. Also, many nonvolatile memory devices have limited operating speeds.

Conveniently, a nonvolatile, high-speed memory device with extended endurance has been developed by combining the SRAM and nonvolatile technologies. Specifically, nonvolatile static random access memory (NVSRAM) devices use a standard SRAM front end that provides high-speed access with near-infinite endurance.[3] However, in the event of a power loss, an external capacitor supplies the required energy for all the SRAM data to be transferred to nonvolatile memory. When power to the memory device is reestablished, the nonvolatile data is transparently retransferred to the SRAM front end for readout or editing.

The NVSRAM device chosen for this hardware platform is the Cypress Semiconductor CY14B101P. This device provides 1 Mbit of storage space configured as 128K addressable locations of 8-bit sized data. As discussed in Section 2.1, each sampling of the linear array generates 128 bytes of data. Therefore, the CY14B101P can store on the order of 1,000 complete samplings of the linear array. Further, with the linear array's sampling rate of approximately 10 kHz, this corresponds to roughly 0.1 s of data logging. This sampling time is more than sufficient for most applications.

Figure 3 displays a pin out diagram of the CY14B101P memory device. Pin names that are grayed-out in this diagram are associated with a real-time-clock feature of this device, which is not utilized. The CY14B101P utilizes a serial port interface (SPI) communication protocol.[4] Similar to the serial format of the linear array, the SPI port incorporates a serial clock line (SCK); a serial data in line, serial input (SI); and a serial data out line, serial output (SO). However, the SPI protocol includes an additional chip select line, chip select (CS_), which provides the potential for a single master device to communicate with multiple servant SPI devices. Furthermore, the SPI protocol drops the use of start and stop pulses, which yields increased data throughput. In contrast to the universal USART format, the SPI protocol transfers data beginning with the most significant bit and proceeding to the least significant bit.

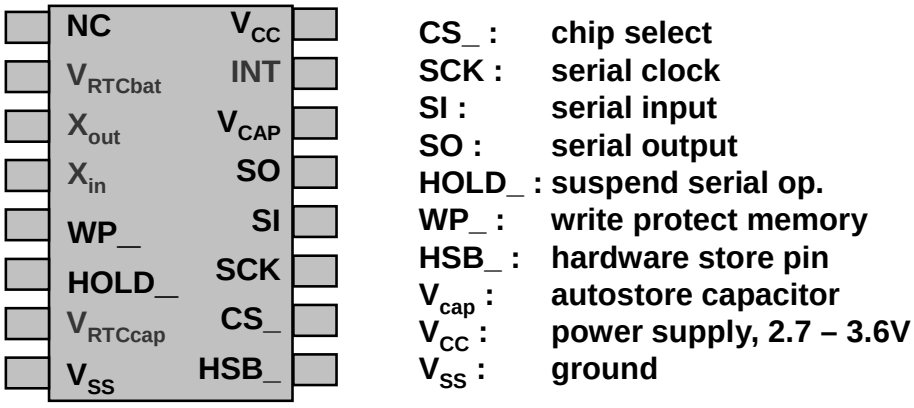

Fig. 3 Pin out diagram of the CY14B101P memory device

SPI communications support multiple operating modes, which are differentiated by the quiescent state of the SCK clock signal. The CY14B101P supports SPI communication modes 0 and 3.[5]

Relationships between CS_, SCK, and data signals for mode 0 and mode 3 SPI operation are illustrated for data input in Fig. 4 and data output in Fig. 5. The mode of operation is selected by the state of the SCK line when the CS_ line goes low.

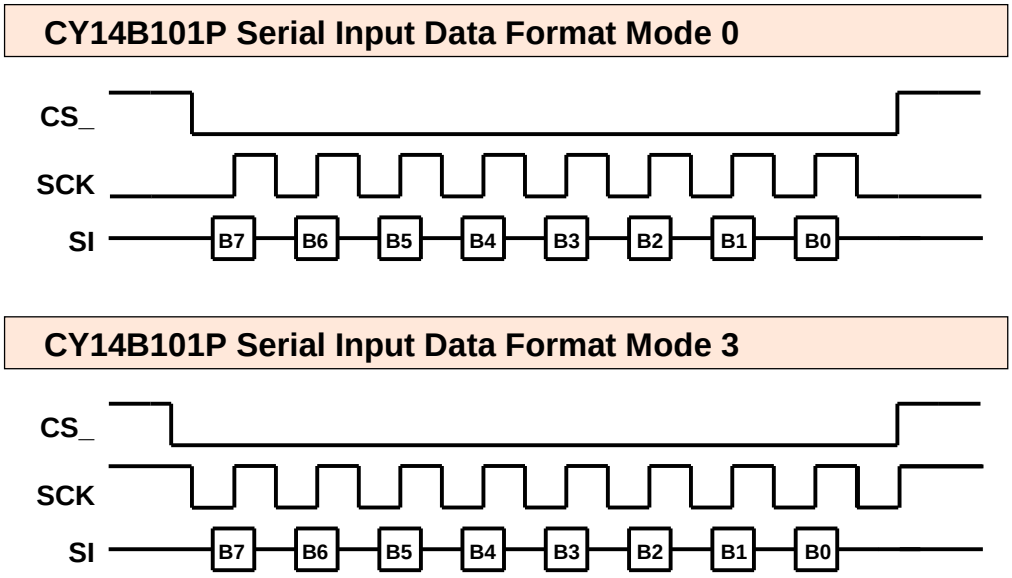

Fig. 4 CY14B101P memory serial input data formats

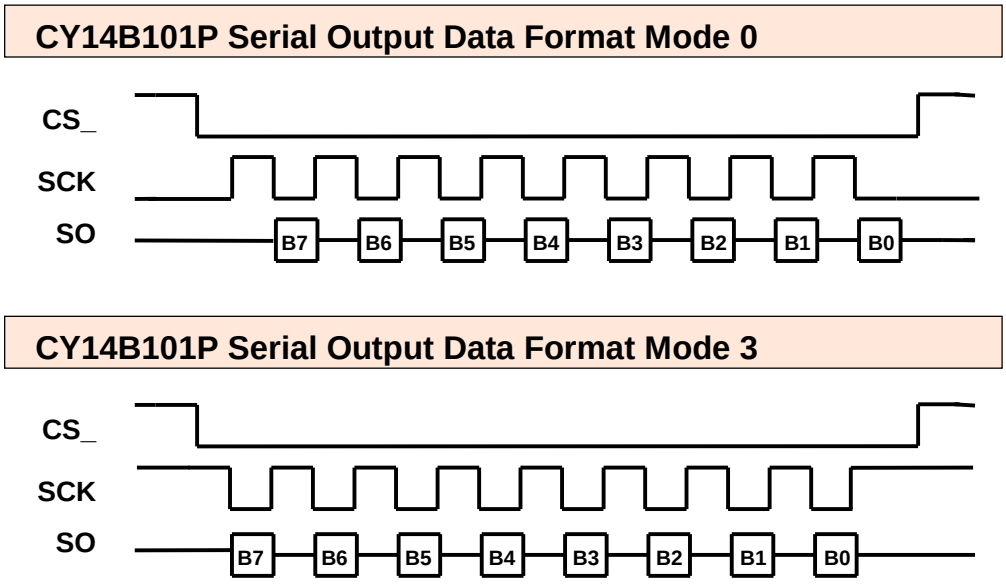

Fig. 5 CY14B101P memory serial output data format

The CY14B101P NVSRAM memory device can operate at clock frequencies up to 40 MHz. This operating speed is more than sufficient to handle the 10-MHz data transfer rates from the linear array. This speed will be a consideration in choosing the specific device for the next major hardware category—the microcontroller.

2.3 Microcontroller

The microcontroller acts as the nexus that links together the multiple subsystems contained in the hardware platform. Serving as the master device, it not only orchestrates the activities of the linear array and memory servant devices, but it also coordinates the timing and communications of all the other hardware components. Many different types of microcontrollers are available. The selection of a suitable microcontroller is based on multiple specific criteria. However, underlying this selection process is the desire to have a microcontroller that offers flexibility, expandability, and ease of use. The desire for flexibility necessitates a microcontroller that can be easily reprogrammed. To ensure expandability, a microcontroller should be selected that offers extensive data and program memory, supports multiple communication protocols, and allows a wide variety of integrated peripherals. For ease of use, the microcontroller must be supported by a mature integrated development environment (IDE) that seamlessly supports mixed languages, code text editing, machine code compilation, and device programming.

A suitable microcontroller for this hardware platform is the Microchip PIC24HJ256GP210A.[6] An onboard in-circuit serial programmer (ICSP) port allows this device to be conveniently reprogrammed at any time. Microchip offers its MPLABX IDE[7] as a free download. This IDE provides a text editor, source-level debugger, project manager, software simulator, and programmer support to assist with code development and microcontroller implementation. Additionally, Microchip offers free downloads of its entry level MPLABXC 16-bit C compiler.[8] The PIC24HJ256GP210A is a high-end 16-bit microcontroller with an extensive supply of program memory and peripheral functionality to allow this hardware platform to be developed well beyond the scope of the current project.

Considering some of the specifics of the PIC24HJ256GP210A, this microcontroller can operate at a central processor unit (CPU) speed of up to 40 million instructions per second (MIPS). This processing speed matches the 40-MHz clock frequency of the CY14B101P NVSRAM memory device. Among the many peripheral devices included in the PIC24HJ256GP210A are USART ports, SPI ports, general purpose input/output (GPIO) ports, a multitude of timers, and direct memory access (DMA) support. The USART ports offer a potential means for communicating with the TSL3301-LF linear optical sensor array, which as noted supports the USART protocol. Similarly, the SPI ports are available for communication with the CY14B101P memory device. The GPIO ports offer another possibility for communicating with these devices where software techniques are employed to realize serial communication rather than dedicated hardware. This form of communication is often referred to as “bit-banging”.[9] Finally, the timers and DMA

support provide the potential for achieving the goal of simultaneous data logging and data processing by running simultaneous, parallel operations. Figure 6 displays the pin out for this 100-pin device and underscores the availability of a wide range of functionality.

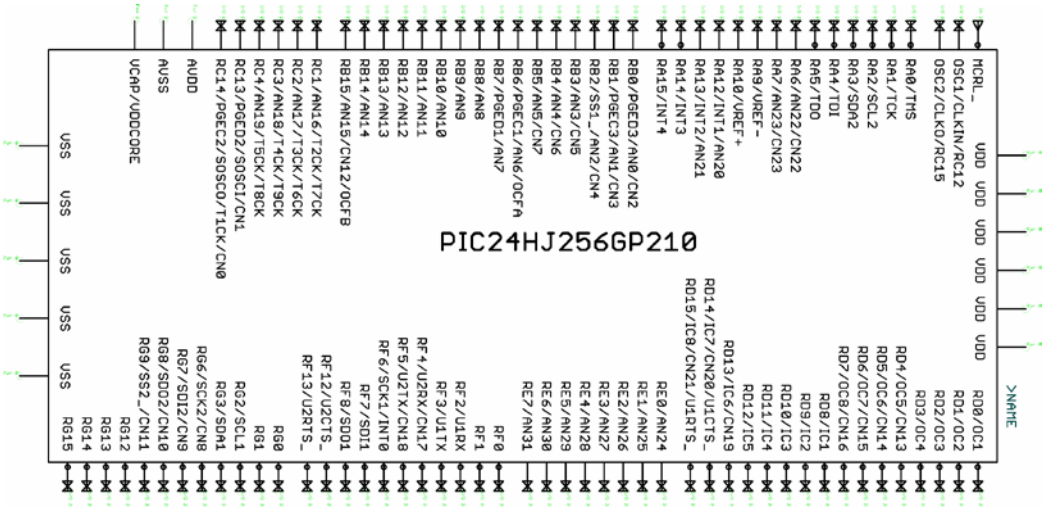

Fig. 6 Pin out for the PIC24HJ256GP210A microcontroller

2.4 Oscillator

An oscillator is needed to generate the timing signal required by the optical linear array, memory, and microcontroller. A wide variety of suitable oscillators are commercially available. The device chosen for this application is the ECS 3963-100-BN-TR.[10] As noted, the maximum clock frequency of the optical linear array is 10 MHz, while the maximum clock frequency for the memory device and the microcontroller is 40 MHz. When recording data from the linear array, the memory device does not need to operate faster than the linear array, so they both can share a common 10-MHz clock signal. Ideally, the microcontroller should operate at its maximum allowed system clock frequency of 40 MHz. Conveniently, the chosen microcontroller provides a phase-locked loop (PPL)[11] capability that allows higher frequency microcontroller system clock signals to be generated from lower frequency time bases.[12] Thus, a single 10-MHz time base can be utilized to clock these three major components at their optimal speeds.

3. Major Hardware Component Integration

Now that the 3 major hardware components—the optical linear array, the memory device, and the microcontroller—have been selected and characterized, the architecture for integrating these devices into an efficient and flexible system can be addressed. Such a system must meet the following list of functional requirements.

a) 10-MHz oscillator clock signals must be available to the microcontroller at all times.

b) The microcontroller must be able to drive the optical linear array serial clock input line via an SPI module and a bit-banger module for device configuration, device interrogation, and data acquisition.

c) The optical linear array serial clock input line must have direct access to the 10-MHz clock signal during free-running data acquisition operations (operations that do not involve microcontroller control or intervention).

d) The optical linear array serial clock input line must be provided to a microcontroller timer module to control the duration of free-running data acquisition operations.

e) The microcontroller must be able to drive the nonvolatile static memory serial clock input line via an SPI module and a bit-banger module for device configuration, data storage, and data retrieval.

f) The nonvolatile static memory serial clock input line must have direct access to the 10-MHz clock signal during free-running data acquisition.

g) The nonvolatile static memory serial clock input line must be provided to a microcontroller timer module to control the duration of free-running data acquisition operations.

h) The microcontroller must be able to drive the optical linear array data input line via an SPI module and a bit-banger module for device configuration, device interrogation, and data acquisition.

i) The microcontroller must have access to the optical linear array data output line via an SPI module and a bit-banger module for device interrogation and data acquisition.

j) The microcontroller must be able to drive the nonvolatile static memory data input line via an SPI module and a bit-banger module for device configuration and data storage.

k) The microcontroller must have access to the nonvolatile static memory data output line via an SPI module and a bit-banger module for data retrieval.

l) The nonvolatile static memory data input line must have access to the optical linear array data output line during free-running data acquisition.

Development of a model of the optical linear array hardware platform is begun by considering the necessary clock signals, as described in items *a* through *g* in the above list of functional requirements. Figure 7 illustrates the clock signal paths between the major hardware components that satisfy these requirements. The specific requirement that each connection satisfies is specified by the corresponding list item index shown in red.

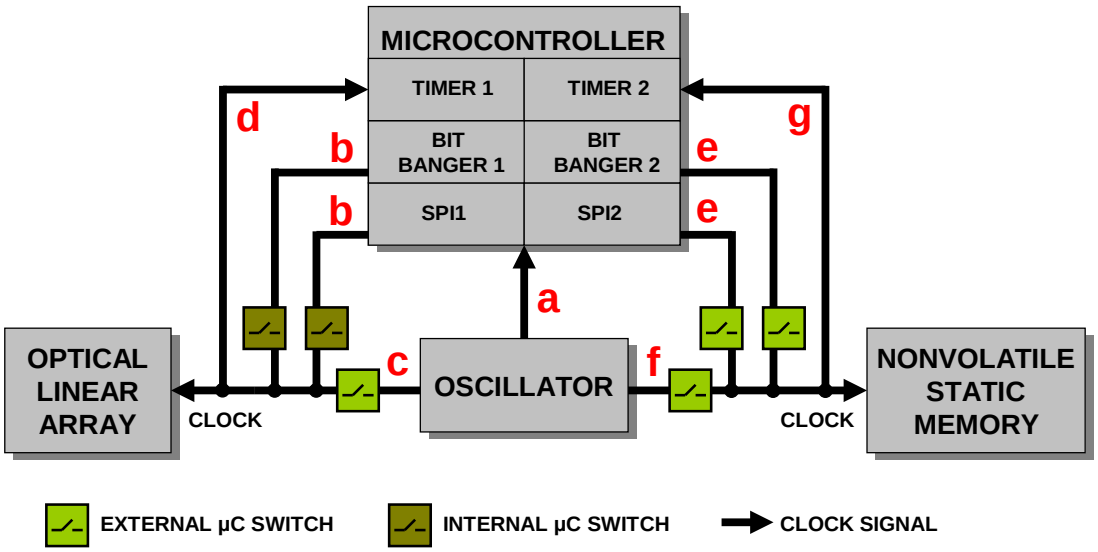

Fig. 7 Clock signal connections between hardware platform major components

From Fig. 7 it is clear that multiple outputs are available to drive the clock input of both the optical linear array and the nonvolatile static memory. In everyday lives, an individual cannot effectively listen to a room full of conversations. For clear communication, that individual can only listen to a single person at a time. Similarly, a digital input can only listen to one digital output at a time. In the clocking networks of Fig. 7, this requirement is satisfied by including switches that, if configured correctly, will only allow the desired clock output source to drive each clock input. Some of this switching capability is provided by the microcontroller itself in the form of digital outputs that are 3-state, or tri-state, capable.[13] Such outputs can be set to either a level 0, a level 1, or a high-impedance state that effectively disconnects the output from external circuitry. Additional external switches are utilized as required. The state of these external switches is also controlled by digital outputs from the microcontroller. In Fig. 7 the digital microcontroller connections to these switches are omitted for clarity.

Developing the model of the hardware platform continues with considering the data signals that must pass between the major components. Figure 8 illustrates these data signal paths. Again, the specific requirement that each data connection satisfies is labeled by the corresponding item index from the previous list. Switches under the control of the microcontroller are once again used to provide a means for selecting a single data output source for each data input. For clarity, the previously discussed clock signals have been omitted.

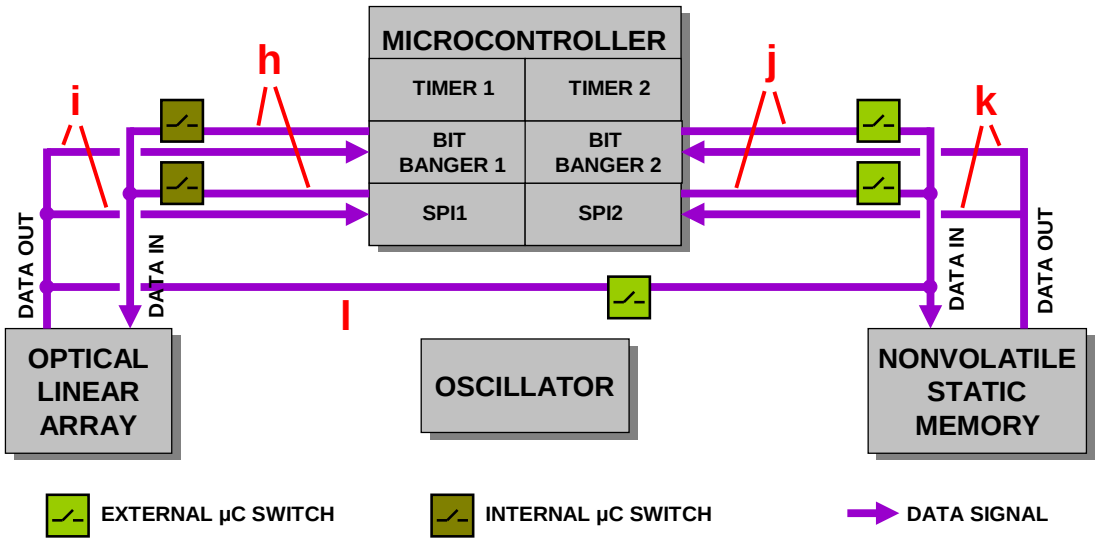

Fig. 8 Data signal connections between hardware platform major components

4. Ancillary Hardware Components

In addition to these major components, there are multiple supporting hardware subsystems that allow the major components to perform their required tasks. These ancillary modules perform a range of functions, including power management, firmware maintenance, communication, and control. Figure 9 displays the major components and their clock and data connections along with the various ancillary components. The appendix provides electronic schematics for the entire optical linear array hardware platform with explicit connections between all major and secondary components. Each of these supporting components is highlighted in Fig. A-1 of the Appendix. Their functions and interactions with the microcontroller are discussed in Sections 4.1 through 4.9.

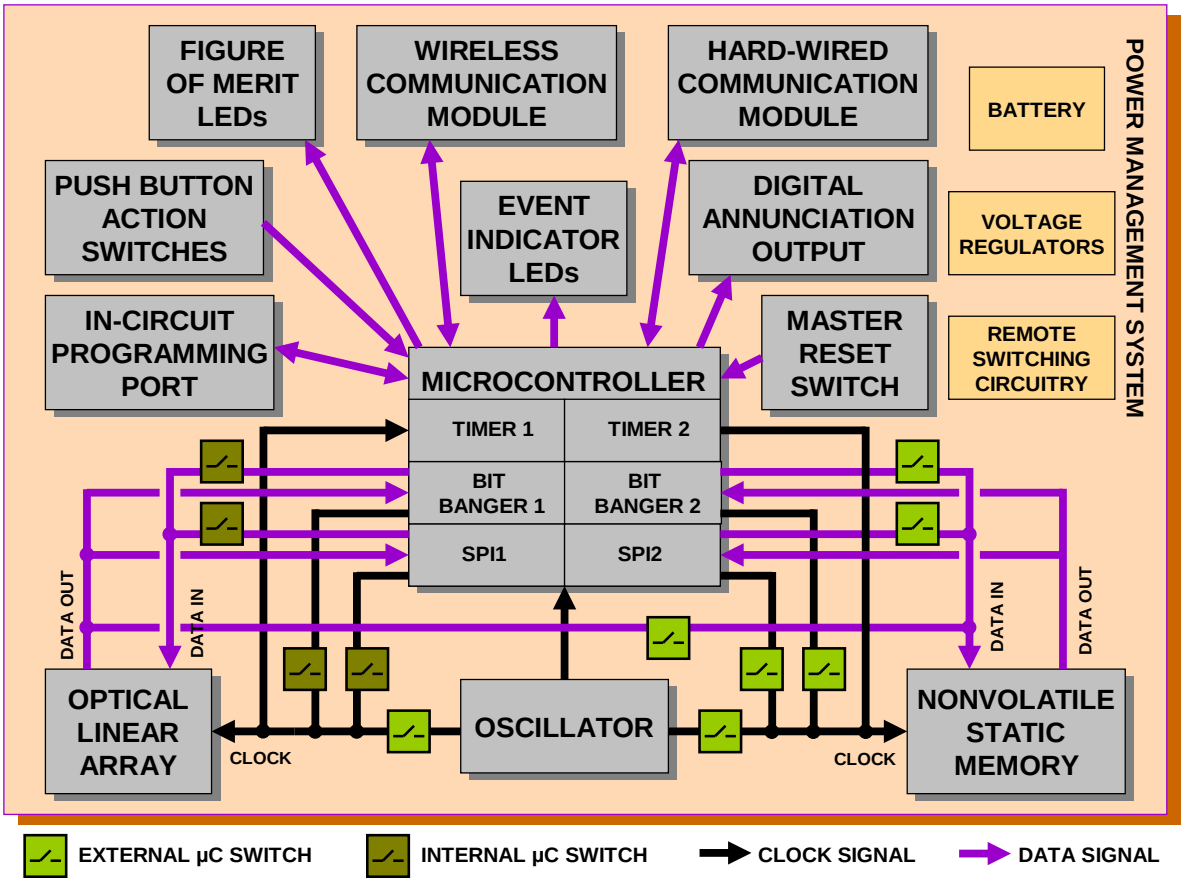

Fig. 9 Block diagram of optical linear array hardware platform

4.1 Microcontroller In-Circuit Serial Programming Port

The ICSP port[14] is used to provide the microcontroller with both programming and diagnostic functionality. For programming duties, the ISCP port makes primary use of 2 microcontroller pins—PGD: a bidirectional serial data line that allows programming data to be read in and device information to be read out on command; and PGC: a timing pin that clocks in data and synchronizes the entire process. In addition to these core pins, the ICSP interface requires connections to the microcontroller's master reset pin, MCLR_, and the hardware platform's common power and ground connections.

The ICSP port also provides a connection for in-circuit debugging. With an appropriate hardware interface and software support, the microcontroller can be single-stepped through code, software breakpoints can be set, and the contents of registers can be monitored. These capabilities are significant assets during debugging operations.

4.2 Master Reset Switch

The master reset operation is familiar to most computer users. When a computer hangs up or presents the "blue screen of death" (as computer freeze is known in popular culture) a common

first aid response is to turn the power to the device off and back on—the classic master reset. Code development activities on this hardware platform will inevitably produce similar scenarios where the system becomes unresponsive and simply goes away to some unknown digital dead end. Toggling the power to the hardware platform serves to reset the microcontroller to a known, initial configuration. The master reset push-button switch provides a convenient means for performing this microcontroller reset without powering down the entire platform or even powering down the microcontroller. In addition to an actual switch, components are included to debounce its operation[15] and to provide immunity to stray static charges.

4.3 Push-Button Action Switches

Two push-button switches on the hardware platform are routed directly to input pins on the microcontroller. These input signal devices are available to the programmer to trigger desired microcontroller software actions, such as calibration activities, data acquisition, data transfer, or debugging events. Passive components are included with these push-buttons to provide switch debouncing.

4.4 Event Indicator Light-Emitting Diodes (LEDs)

Converse to the push-button action switches, 2 microcontroller output pins are routed directly to LEDs on the hardware platform. These optical signaling components can be utilized by the microcontroller programmer to denote the initiation or completion of specific events, the status of individual bits, or the requests for user actions.

4.5 Figure of Merit LEDs

The 16 output pins from one of the microcontroller's registers are routed to 16 independent LEDs. Thus, any 16-bit number that is loaded into this register can be visually displayed on the hardware platform. This arrangement provides the opportunity to easily display the magnitude of any sensor input value, calibration value, calculation value, threshold value, etc., as a numerical value or as a representative analog signal strength value. For scenarios where this magnitude display capability is not required, these 16 LEDs can be used as independent digital status bit indicators.

4.6 Digital Annunciation Output

This hardware platform was developed to explore adaptive fuze concepts. Fuzes are generally employed to activate an associated reactive device at an appropriate time or condition. Therefore, this hardware platform includes a dedicated digital output that is available to initiate associated components.

4.7 Wireless Communications Link

A communications link is necessary to provide a means for transferring or downloading recorded data from this hardware platform for long-term storage and analysis. One particularly convenient

method for communicating with the platform is via a wireless link. By obviating the need for a physical connection, data can be transferred in situ following a ballistic event. This capability provides a potentially significant safety advantage for experimental scenarios where the post-event environment may be hazardous. In addition, the wireless communication link can be used to dynamically reconfigure the hardware platform and monitor its status.

The wireless communications link for the hardware platform is based on a Linx Technologies TRM-418-LT transceiver module[16] and a Splatch quarter-wave antenna chip.[17] This combination provides reliable, bidirectional serial data communication over distances of dozens of meters at baud rates up to 10,000 bits per second (bps). Compared to the 10-megabits bps (Mbps) data rate of the TSL3301-LF optical sensor, this baud rate is quite modest. Clearly, real-time data transfer directly from the optical sensor is not a possibility over this wireless link. However, post-event data transfer from the CY14B101P memory device is certainly possible.

The TRM-418-LT transceiver module is designed to be as transparent as possible. Toward this end, the transceiver does not perform any internal encoding or decoding associated with a particular communications protocol but simply transmits radio signals that are exact copies of the digital, serial signal input. Conversely, when receiving data, the transceiver provides a digital serial signal that is an exact copy of the wireless transmission. Communications to and from the TRM-418-LT transceiver module are routed through one of the microcontroller's two USART serial ports.

4.8 Hard-Wire Communications Link

Regardless of the convenience of the wireless communications link, there may be experimental scenarios where radio frequency communications are not permitted because of safety considerations. For these situations, a conventional hard-wire communications link is provided. A schematic circuit for this communications link is displayed in Fig. A-2 of the Appendix. The core of this link is a Maxim Integrated Products MAX3232 RS-232 transceiver.[18] This device converts the unipolar USART signals from the microcontroller to inverted bipolar signals for direct communication with a conventional computer serial port.

4.9 Power Management System

Perhaps the least glamorous module, but certainly one of the most important modules, of the optical linear array hardware platform is the power management system. This importance is connoted by the fact that this system encompasses and contains all the other components in the block diagram of Fig. 9. Clean, reliable power is required by all components. As displayed in the schematic diagram in Fig. A-3 of the Appendix, the power management system includes 3 primary elements. A lithium polymer battery provides power for stand-alone operations. With 2 cells connected in series, this battery provides a nominal 7.4 volts (V). The second element of the power system is the voltage regulation circuitry that knocks the 7.4 V provided by the battery down to the 3.3 V that all components on the hardware platform are designed to utilize. There

are, in fact, 2 separate voltage regulator networks—one that provides 3.3-V power for the digital components and one that provides 3.3-V power for the analog components in the microcontroller. Digital components tend to generate voltage glitches and spikes that can be transferred through the power connections. By using an independent analog power network, this source of digital noise is effectively isolated from the analog components.

The final element of the power management system is a collection of connectors and switching circuitry that provide considerable flexibility for powering the hardware platform. Battery charging terminals are provided for charging the lithium polymer battery with an external voltage and current regulated power supply. Once charged, the hardware platform can be manually activated by inserting a jumper between 2 direct circuit activation header pins. For extended laboratory testing and software development, where it may be inconvenient to rely on the battery for power, the hardware platform can be powered externally, regardless of the charge state of the battery, by hooking a 5- to 15-V power supply to external power terminals.

For ballistic range experiments, safety considerations generally do not allow the hardware platform to be activated during the projectile loading process. Thus, a means for activation of the hardware platform is required after the loading process has been completed. A remote circuit activation switch is provided that consists of field effect transistors (FET) and a resistor-capacitor (RC) driving circuit. An external voltage applied to the remote circuit activation terminals turns on the hardware platform while depositing charge to the capacitor. The hardware platform will remain activated as long as this external voltage is maintained to the remote circuit activation terminals. When the external voltage is removed, the charge stored in the capacitor will continue to keep the FETs and the hardware platform activated. The length of time the capacitor will keep the hardware platform turned on is determined by the resistor in the RC circuit, which is simultaneously draining the charge from the capacitor. For the RC component values shown in Fig. A-3, this operational time is nominally 20 s. Longer, or indefinite, activation times can be realized using the microcontroller output line, which is also connected to the remote circuit activation switch input line. A high level generated by this digital microcontroller line can serve as a feedback connection to maintain the charge state of the capacitor and, thus, the battery powered operation of the hardware platform.

5. Optical Linear Array Hardware Platform Embodiment

The optical linear array hardware platform is assembled on a custom printed circuit board (PCB) that is circular in shape with a diameter of 3.6 inches. Figures 10 and 11 illustrate the design of the multiple layers that are included in the PCB. In addition to the 4 conductive copper signal carrying layers, there are 2 solder stop layers that facilitate the soldering of components to the board and 2 silkscreen layers that indicate the position and identity of the electronic components. The order of the physical layers from top to bottom is top silkscreen, top solder stop, top copper,

insulation, layer 2 copper, insulation, layer 3 copper, insulation, bottom copper, bottom solder stop, and bottom silkscreen.

The top side of the PCB contains the majority of the digital and power management components. The bottom of the PCB, which presumably would be oriented to face in the direction of the projectile shot line, contains the optical linear array, the figure of merit LEDs, the push-button action switches, and the wireless communication components. An array of 8 mounting holes is evenly spaced around the outer edge of the PCB. These mounting holes are sized as pass-holes for 6-32 screws and threaded rod. Three more holes arranged in a triangular pattern near the center of the PCB are available for mounting lens systems in front of the optical linear array. These lens mounting holes are sized as pass-holes for 2-56 screws.

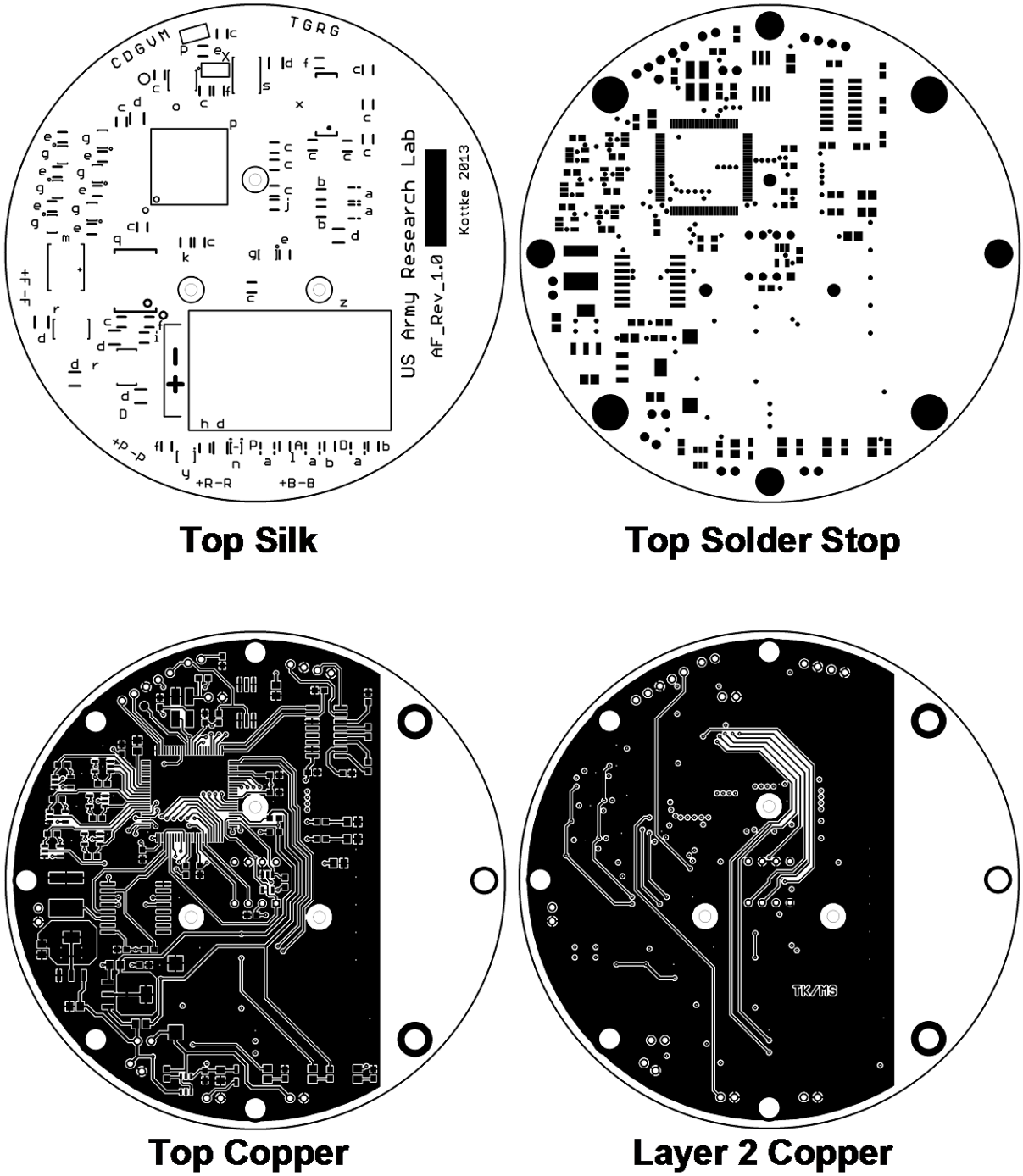

Fig. 10 Design of hardware platform printed circuit board upper layers

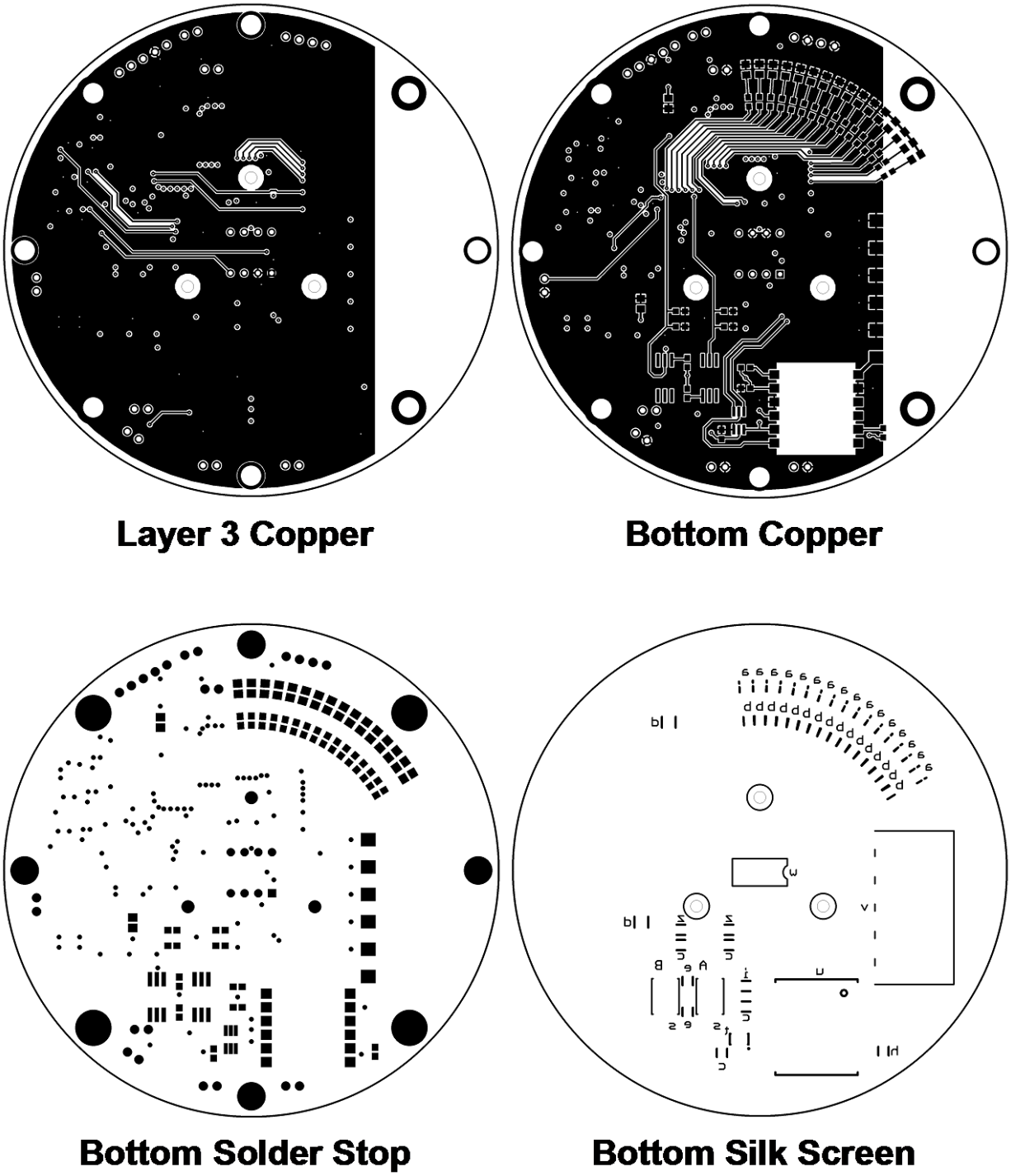

Fig. 11 Design of hardware platform printed circuit board lower layers

Figure 12 displays the top and bottom views of a completed linear optic array hardware platform PCB along with the locations of the various components and modules. The weight of the hardware platform as illustrated in Fig. 12 is less than 50 g and the total thickness is less than 30 mm.

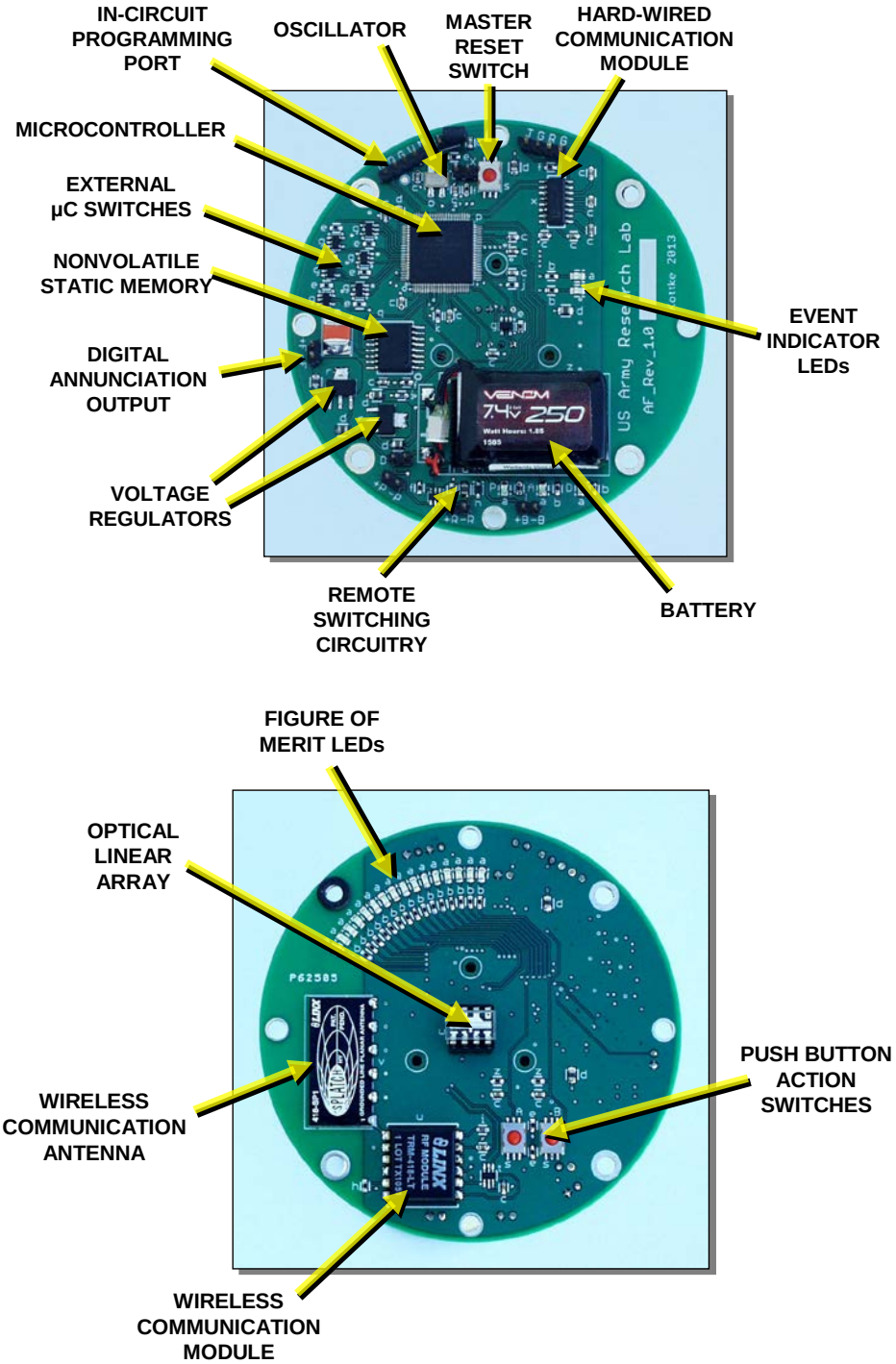

Fig. 12 Top view (above) and bottom view (below) of completed hardware platform printed circuit board

6. Optical Linear Array Hardware Platform Operation

6.1 Optical Linear Array Data Acquisition

Data from the optical linear array are routed to a special section of the microcontroller's memory that contains dual-ported SRAM (DPSRAM). This type of memory allows a type of data transfer known as DMA.[19] Within the DPSRAM space, both the CPU and an onboard DMA controller can write to and read from common memory locations with no interference. Furthermore, the DMA's independent data bus and address structure allow the transfer of data between certain microcontroller peripherals, such as the SPI and the USART, and the microcontroller's onboard memory with no impact on CPU operations. This parallel bus architecture means data can be transferred to and from these peripheral devices while the CPU is performing other tasks.

Figure 13 illustrates how the DPSRAM is organized. All the data memory for the PIC24HJ256GP210A microcontroller is sized as 16-bit words. The DPSRAM memory on this microcontroller begins at hexadecimal address 4000h. Word addresses must always be even numbers. So the next word-sized memory addresses are 4002h, 4004h, etc. Data memory can also be accessed in byte-sized units with each data word consisting of a more significant byte (MSByte) containing bit locations 8 through 15 and a less significant byte (LSByte) consisting of bit locations 0 through 7. Byte addresses can be both even and odd.

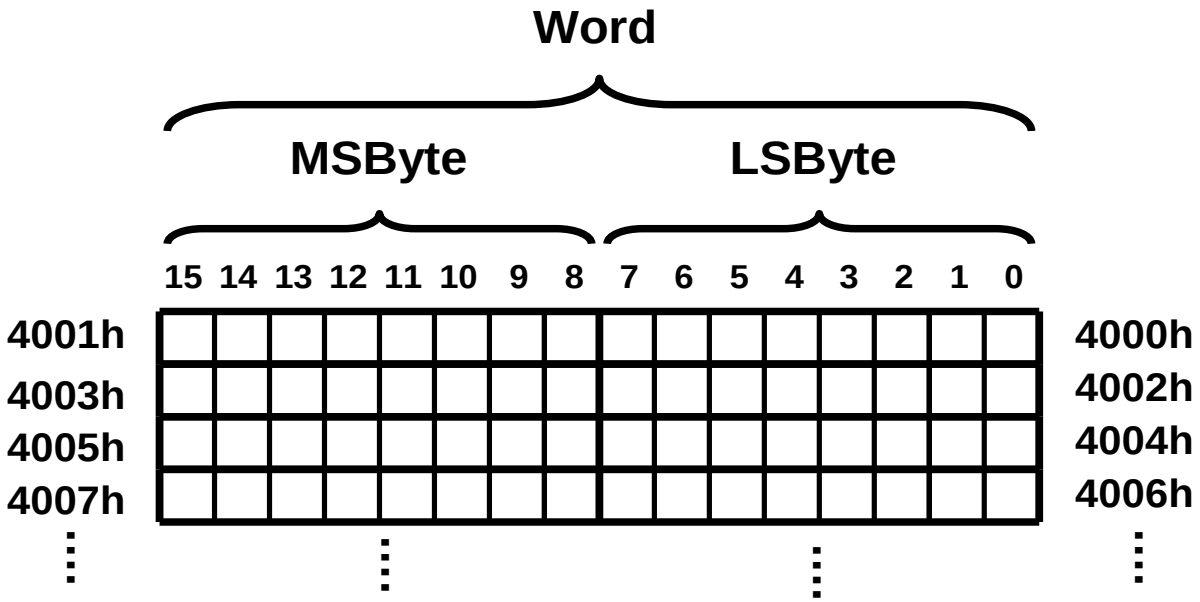

Fig. 13 Organization of DPSRAM in PIC24HJ256GP210A microcontroller

Recall from Fig. 2 that data bytes from the optical linear array are output serially beginning with a start bit, followed by 8 data bits proceeding from the least significant to the most significant bit, and ending with a stop bit. This 10-bit data transfer format is most unfortunate because it does not fit efficiently into either a 16-bit word memory format or in an 8-bit byte memory format. As a result, transferred optical linear array data is split across word and byte boundaries, is sprinkled with start and stop bits, and exhibits an unconventional bit-wise endianness.[20] What follows is an explanation of the specifics of the data transfer process and the procedure for reformatting these data into a usable form.

DMA byte transfers from SPI peripherals deposit bit values into the least significant bit of the designated byte address and left-shift these values as additional bits are transferred. Once a byte location is completely filled, the next data bit is deposited to the least significant bit of the following byte address location. Figure 14 displays how this DMA SPI data transfer precedes for the optical linear array data filling the first data word. In row a) of this figure, the start bit, designated as a green block, has been transferred to the least significant bit of the lowest DPSRAM location. The next bit to be transferred is the bit 0 value of pixel 0. This bit is designated as 0,0 where the first number represents the pixel number and the second number represents the data bit location. Row b) shows how, as the 0,0 bit value is transferred to the least significant bit location, the start bit value is left- shifted to bit location 1. Similarly, rows c) and d) illustrate bit values 0,1 and 0,2 being deposited into the least significant bit position while the preceding bit values are left -shifted. This process continues until, with the transfer of bit value 0,6, the first byte location is completely filled, as seen in row e). In row f) the bit value, 0,7, has been located in the least significant bit of the next byte location, 4001h. Because this bit is the most significant bit of this pixel value, in row g) the next bit to be transferred is the stop bit designated as a red block. At this point, the data from pixel 0 has been completely transferred. Pixel 1 data transfer proceeds with its start bit as shown in row h) followed by its least significant bit 1,0 as shown in row i). This process continues until the second byte location is completely filled as shown in row j).

The optical linear array contains 102 pixels. Each pixel value is serially transferred by 10 bits—one start bit, 8 data bits, and 1 stop bit. Therefore, the transfer of all 102 pixels worth of data requires 1,020 bits, or 128 bytes, or 64 words of data space. A block of data of this size would extend from DPSRAM address 4000h through 407Fh. Figure 15 illustrates the pattern of transferred data in the beginning of this data space. Note the arrangement of data bits can be seen to repeat after 5 words of data in an A, B, C, D, E, A, B, C, etc., pattern. This pattern of repetition will prove useful when the job of reassembling the serially transferred pixel data into a usable format is tackled.

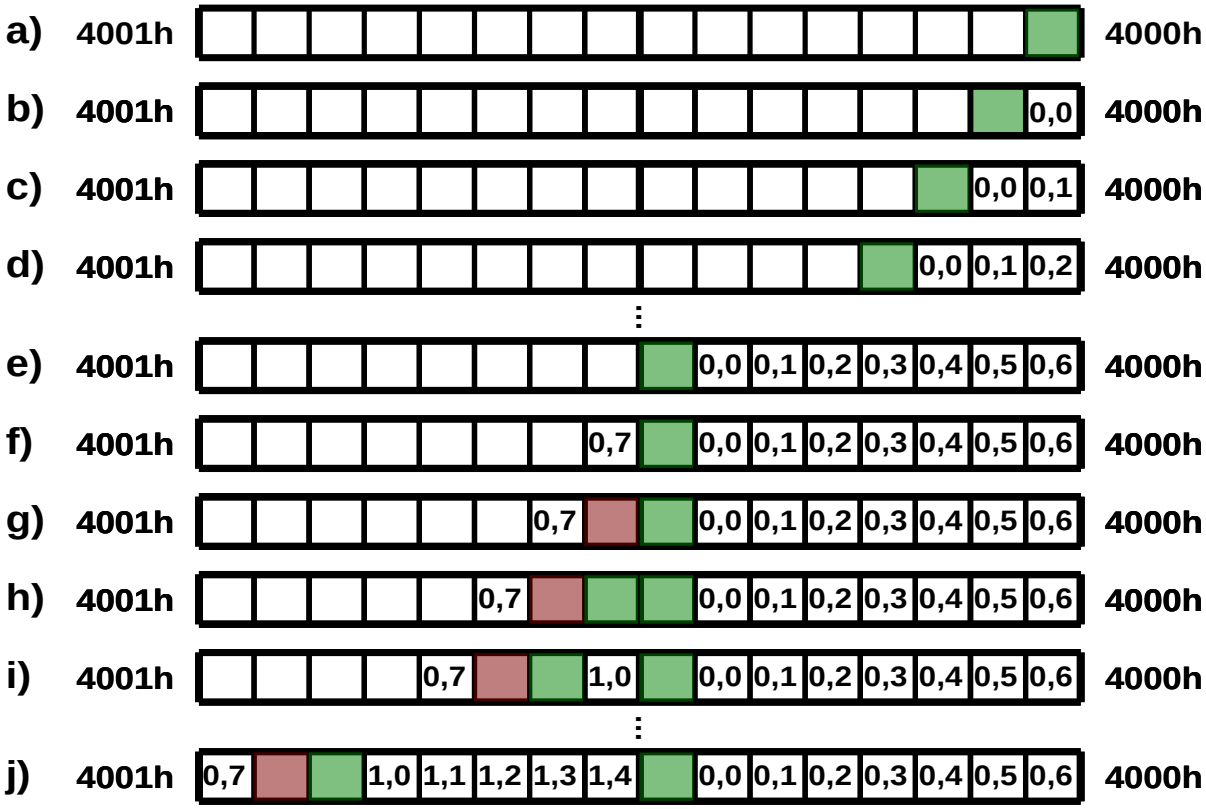

Fig. 14 Illustration of the DMA SPI data transfer process

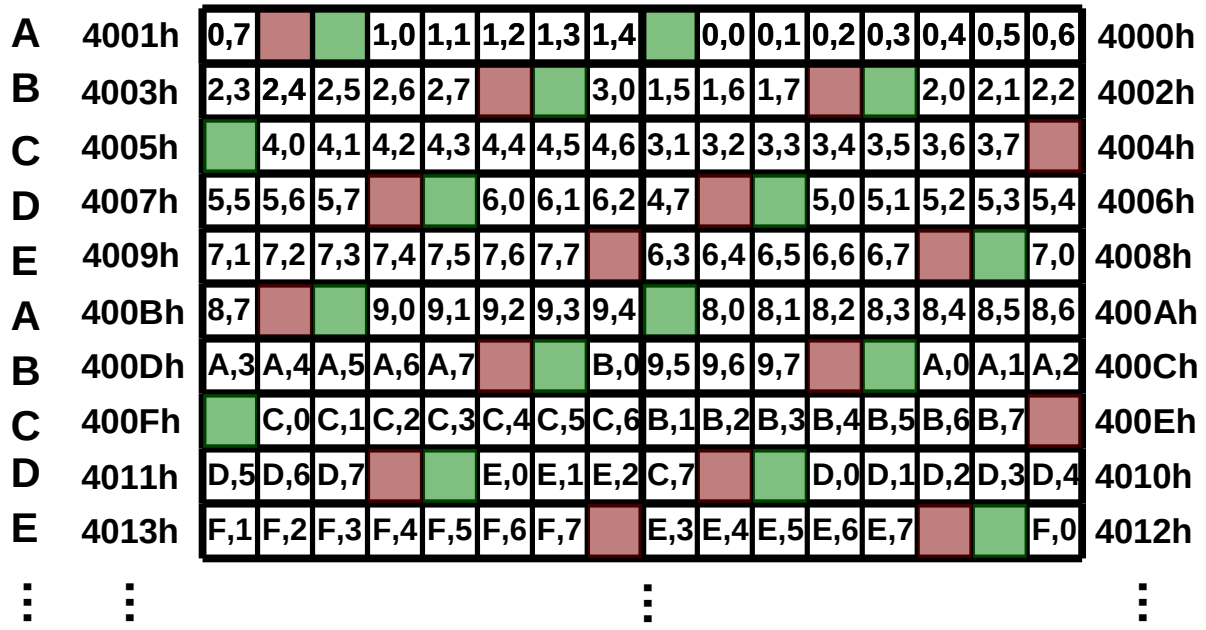

Fig. 15 Format of optical linear array data in DMA data space

Figure 16 illustrates the activity on the optical linear array's serial lines during the data acquisition and transfer process. The upper dark-blue trace shows the activity on the SCLK. Working down on this oscillograph, the light-blue trace shows the SDIN activity where commands are fed into the linear array from the microcontroller. The magenta trace shows the SDOUT activity where data from the linear array are output to the microcontroller. Finally, the green trace is from the microcontroller's digital annunciator output line (DIG OUT) that has been programmed to toggle for specific events. This digital output is convenient for determining operation times and for triggering the oscilloscope.

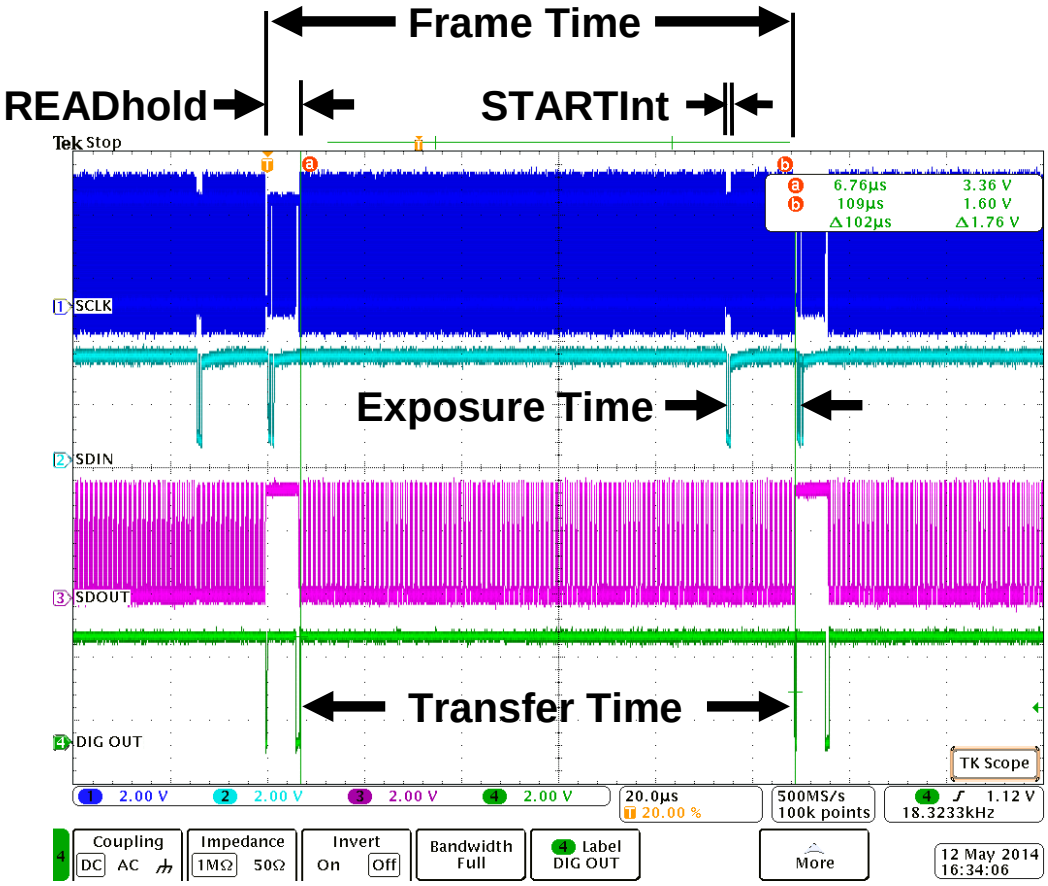

Fig. 16 Signals associated with serial data transfer between optical linear array and microcontroller

The data acquisition and transfer process begins with the microcontroller issuing a READhold command to the optical linear array via the SDIN line. This command initiates multiple activities within the linear array. First the linear array terminates the process of light collection on the optically active pixel elements. Then the light exposure values for each pixel are stored in buffer memory. Finally, the process of serially outputting these buffer memory values is initiated. The issuing of the READhold command is handled by a bit-banger port on the microcontroller. This bit-banger port generates the 10-MHz clock pulses and also toggles the 1-byte READhold

command on to the SDIN line. The individual 10-MHz clock pulses cannot be resolved in this oscillograph. However, the amplitude of the bit-banger generated clock pulses can be seen to be slightly smaller than the surrounding clock pulses. Thus, the temporal extent of the READhold command can be visualized. The short data burst on SDIN at the beginning of the READhold event is the 1-byte command being passed from the microcontroller to the linear array. Additional clock pulses are required to work the READhold command through the linear array's logic circuitry. Therefore, the temporal extent of the READhold command is considerably longer than the time required for the 1-byte SDIN transfer. For clarity, the green DIG OUT signal is also pulsed at the beginning and at the end of the READhold event.

At the completion of the READhold command, the optical linear array has its marching orders so it begins serially transmitting pixel data to the microcontroller using clock pulses from the onboard 10-MHz oscillator. The activity on the magenta SDOUT trace represents the DMA data transfer illustrated in Figs. 14 and 15. Toward the end of this data transfer, the microcontroller generates a STARTInt command to the optical linear array. This command instructs the linear array to begin exposing the optically active pixel elements to light to generate the pixel values that will be transferred during the subsequent frame of data transfer. The timing of the STARTInt command, and thus the duration of the exposure period, is controlled in software.

The total duration of the serial data transfer process is 102 μs. This transfer process takes exactly the amount of time expected for 1,020 bits of data to be transferred by a 10-MHz clock. Note that the generation of the STARTInt command did not extend the data transfer time. The execution of this command does not affect the data transfer time because of the previously noted fact that the optical linear array can simultaneously process data input and data output operations. The overhead added by the generation of the READhold command yields a total frame time of 109 μs. Thus, the hardware platform can acquire optical linear array data at a maximum rate of slightly more than 9,000 frames per second.

6.2 Optical Linear Array Data Formatting and Analysis

Recall that the DMA data transfer between the SPI port and the DPSRAM of the microcontroller proceeds without the services of the microcontroller's CPU. Therefore, except for when the READhold and STARTInt commands are being generated, the CPU is idle. This availability of CPU time is convenient because the transferred data, as illustrated in Fig. 15, is a mess. Noncontiguous and interspersed with start and stop bits, this data is in serious need of reformatting. Happily, the CPU is available to perform this task. An assembly language routine consisting of approximately 3,000 lines of code parses the "raw" data of Fig. 15 into a "cooked" format as illustrated in Fig. 17. With the removal of the start and stop bits, the size of the data block is reduced from 128 bytes to 102 bytes.

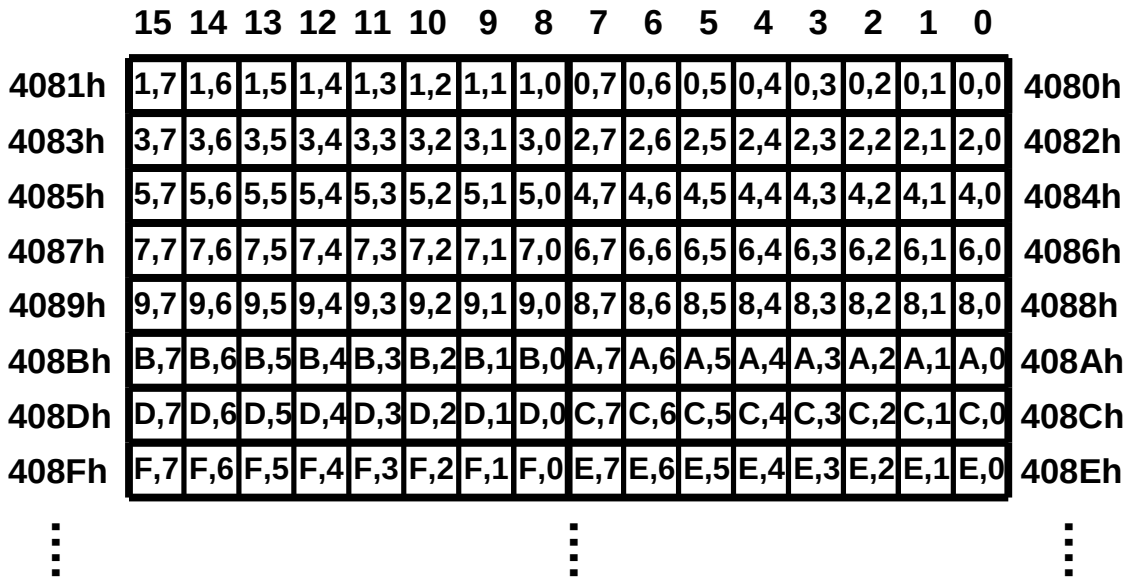

Fig. 17 Arrangement of pixel value data following the reformatting process

This process is simplified by invoking a DMA data transfer option known as ping-pong. With this method, there are actually 2 designated blocks of DMA memory to which linear array pixel data sets are transferred. The transfer of data is alternated between these 2 blocks. So, while current data is being transferred to 1 block, the previous data set in the alternate block can be reformatted without interfering with the data transfer process. Figure 18 shows an example of this process. Pixel data is alternately transferred to DMA block 4000h through 407Fh and 4100h through 417Fh. When not active in the data transfer process, raw data in block 4000h through 407Fh is reformatted and stored into scratch pad locations 4080h through 40E5h. Similarly, when not active, raw data in block 4100h through 417Fh is reformatted and stored into locations 4180h through 41E5h. Although difficult to read, the reformatted data in these scratch pad areas can be seen to represent background ambient light with pixel values of about 07h.

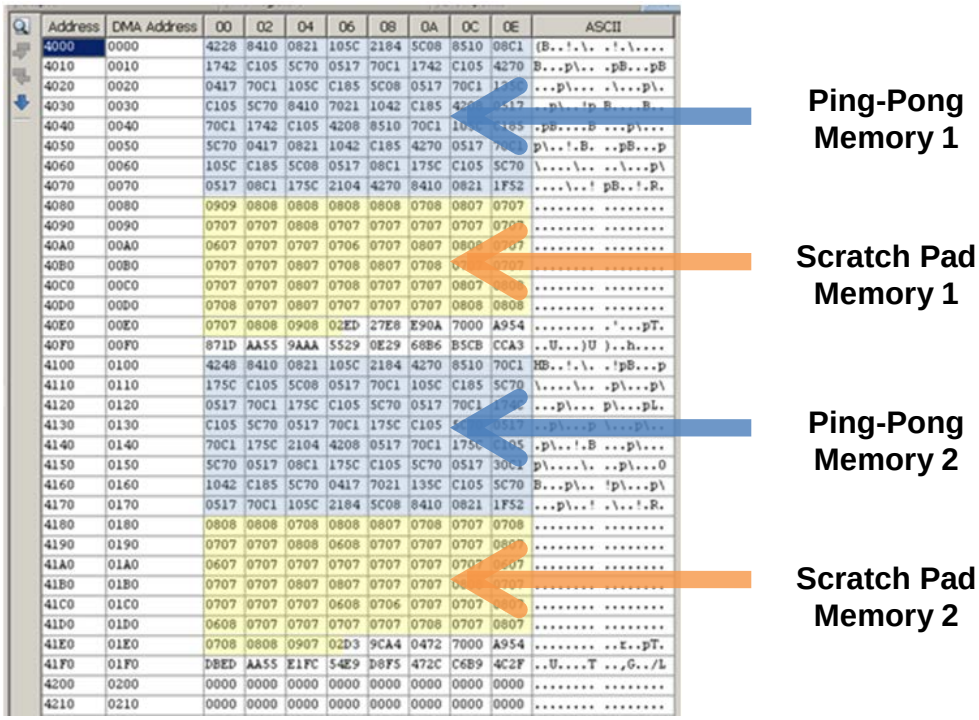

Fig. 18 DMA ping-pong memory blocks and associated scratch pad memory

Figure 19 once again illustrates the signals associated with data transfer between the optical linear array and the microcontroller. However, in this figure, the green digital output trace has been reprogrammed to highlight the CPU usage associated with the data reformatting process. Approximately 70 μs of CPU time are required to reformat the linear array data, which leaves more than 30 μs free for additional data processing and analysis. With the microcontroller's 40- MHz clock speed, this 30 μs time interval provides over 1,200 operational cycles. The PIC24HJ256GP210A microcontroller includes a high-speed 17-bit by 17-bit hardware multiplier that executes in a single cycle and an enhanced arithmetic logic unit that performs divisions in 19 cycles.[21] Therefore, quite a lot of data processing and analysis can be performed in the allotted 30 μS.

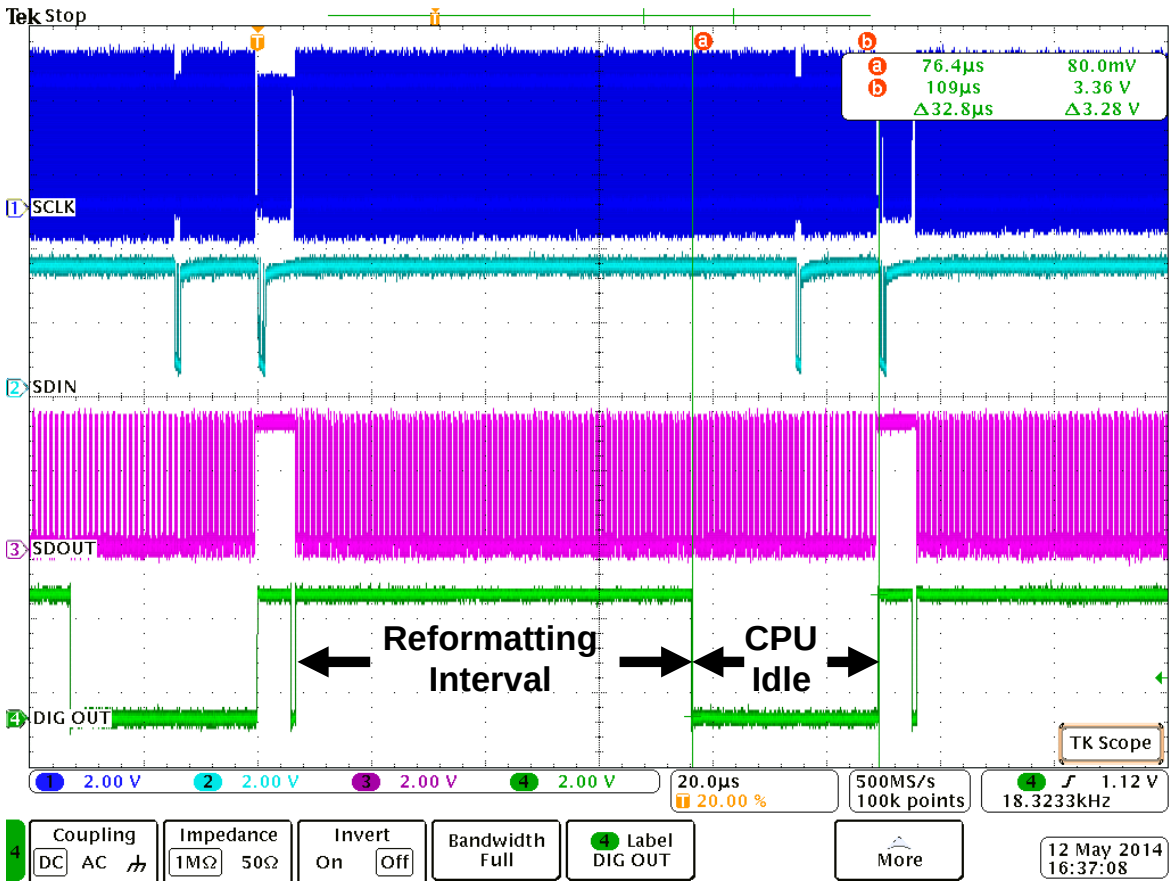

Fig. 19 Serial data transfer between optical linear array and microcontroller displaying CPU usage for data reformatting

6.3 Optical Linear Array Data Monitoring and Display

The DMA data transfer method has been discussed as it relates to the transfer of data from the optical linear array to the microcontroller's onboard memory. In particular, the ability of DMA data transfers to proceed without CPU oversight has been highlighted. There are, in fact, 8 independent channels of DMA transfer available in the PIC24HJ256GP210A microcontroller. This multitude of available background data transfer options has been exploited to increase the functionality of the optical linear array hardware platform.

A second channel of DMA data transfer has been configured to constantly transfer reformatted linear array data from the designated DMA memory space to the onboard wireless communication module via one of the microcontroller's USART ports. An external wireless module receives these data, which are transferred to a Visual Basic program running on a personal computer (PC) for real-time display. Figure 20 shows an example of optical data for the linear array being illuminated through multiple slits. The ability to remotely monitor the output from the optical linear array in real time is useful for lens alignment activities, setting gain levels, and confirming the operating status of the hardware platform.

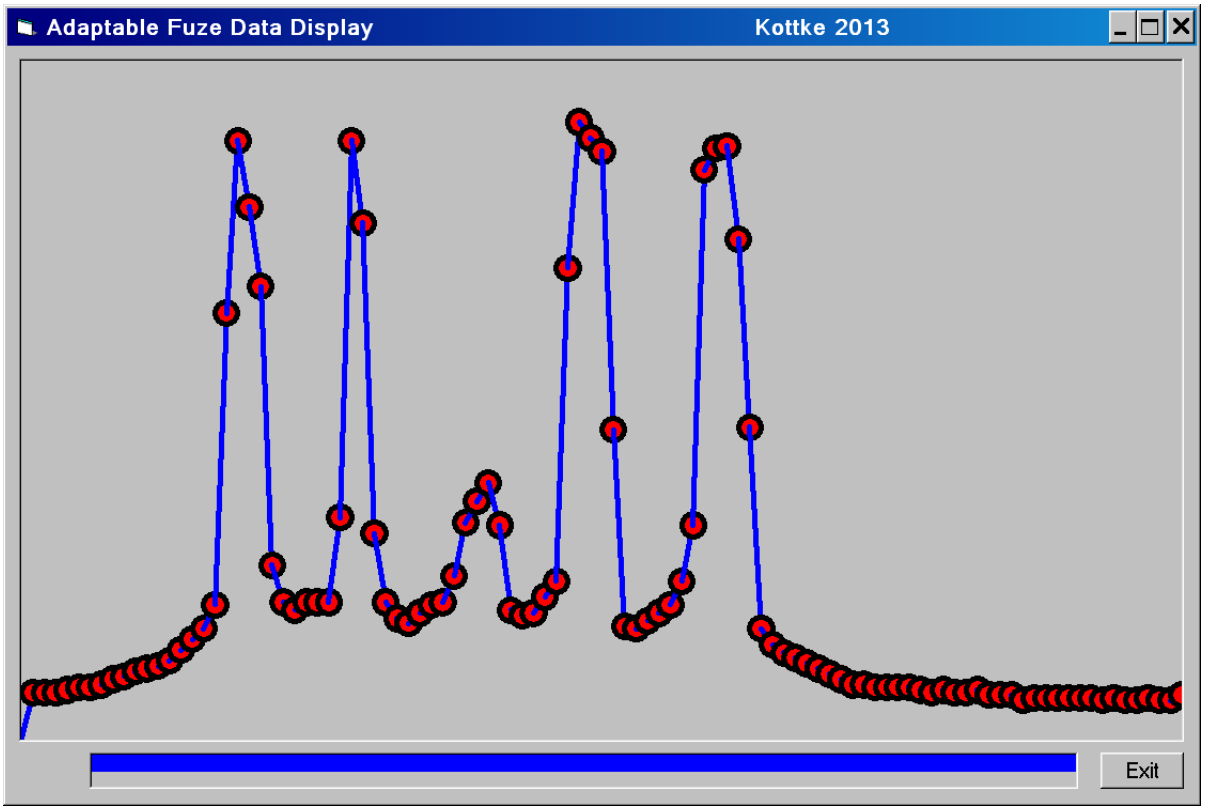

Fig. 20 Example of real-time data monitoring of optical linear array output

6.4 Real-Time Storage of Optical Linear Array Data

One of the desired functions of the optical linear array hardware platform was to serve as a data logger. To satisfy this requirement, reformatted linear array data is constantly transferred from DPSRAM to the external NVSRAM device during the data acquisition process. DMA data transfer through an SPI port is utilized yet again to minimize reliance on valuable CPU clock cycles. The CY14B101P NVSRAM device is configured as a ring buffer so the most recent data constantly overwrites the oldest stored data. This device has an address auto-increment feature that assists with this process. At a desirable time, presumably when analysis of acquired linear array data determines an optimal fuzing event has occurred, this data transfer process is terminated and the NVSRAM device is left holding the optical linear array data leading up to the fuzing event. What remains is for this recorded data to be downloaded for long-term storage and analysis.

6.5 Off-Platform Transfer of Optical Linear Array Data

The determination of a fuzing event terminates the data acquisition process. Optical linear array data is no longer read into the microcontroller, data is no longer reformatted by the CPU, real-time monitoring data is no longer output through a universal asynchronous receiver transmitter (UART) to the wireless communications module, and reformatted data is no longer output through an SPI to the NVSRAM device. After terminating the data acquisition process, the

microcontroller then reconfigures the hardware platform for the transfer of recorded NVSRAM data to an off-platform device, such as to a PC, for long-term storage and post-event analysis. In particular, the NVSRAM device is configured to allow its stored data to be read by the microcontroller CPU through an SPI port. This input data is then transmitted by the wireless communications module via a UART.

6.6 Summary of Optical Linear Array Hardware Platform Data Acquisition and Transfer Operations

The operation of the optical linear array hardware platform has been described, including data acquisition, reformatting, analysis, monitoring, display, short-term storage, and off-platform download. A number of data transfer processes have been discussed involving a variety of microcontroller access ports and external components. To avoid confusion and solidify understanding, a short-hindsight overview is presented.

Consider Fig. 21, which schematically presents the hardware platform during the data acquisition process. Pixel data from the optical linear array are routed to an SPI port on the microcontroller. These data are transferred to a block of DPSRAM memory in the microcontroller using DMA Channel 0, which requires no CPU intervention. As displayed in Fig. 15, these data are in serious need of reformatting. The CPU reads the raw data from this DPSRAM memory block using the conventional data bus, reformats it, and writes it to a different block of DPSRAM memory reserved for reformatted cooked data. From this cooked DPSRAM memory block, multiple DMA data transfer operations are performed simultaneously. Using DMA Channel 1, these data are transferred to a UART connected to the wireless communication module, which allows these data to be monitored and displayed in real time on a PC. DMA Channel 2 transfers cooked data to an SPI port connected to an NVSRAM device that serves as a data logger. Additionally, DMA Channel 3 transfers data back from this SPI port to satisfy a peculiarity of SPI port operation.

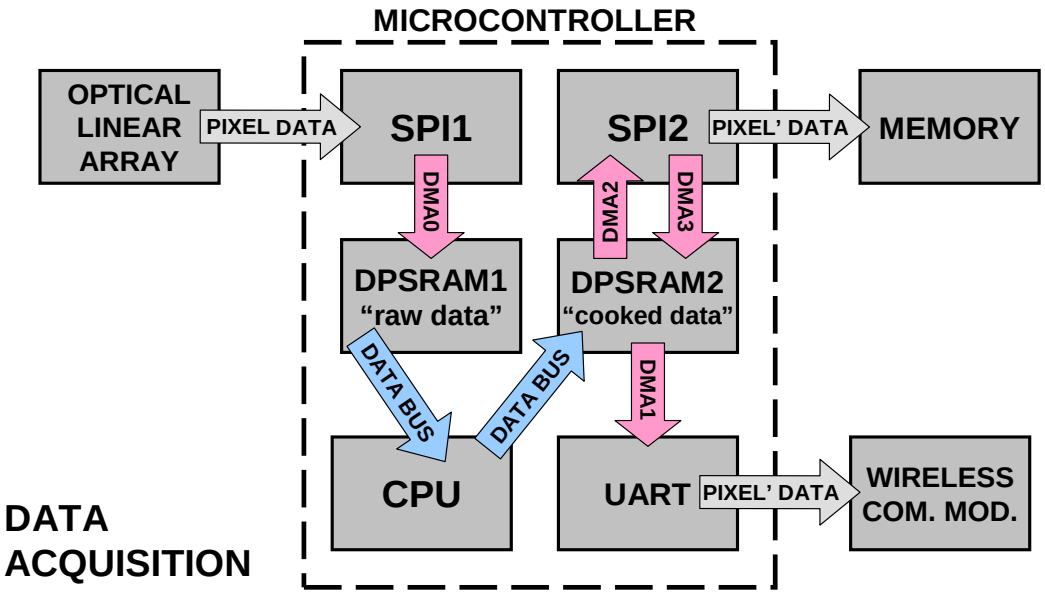

Fig. 21 Schematic diagram of hardware platform operation during data acquisition

When the data acquisition process is terminated, the data transfer operations depicted in Fig. 21 are terminated as well. The microcontroller then reconfigures the hardware platform so data stored in the NVSRAM data logger device can be downloaded to a PC for long-term storage. Figure 22 schematically illustrates the data download process. First, data from the NVSRAM data logger device are read into the CPU through an SPI. These data are then transferred to a UART connected to the wireless communications module that broadcasts this information to a PC.

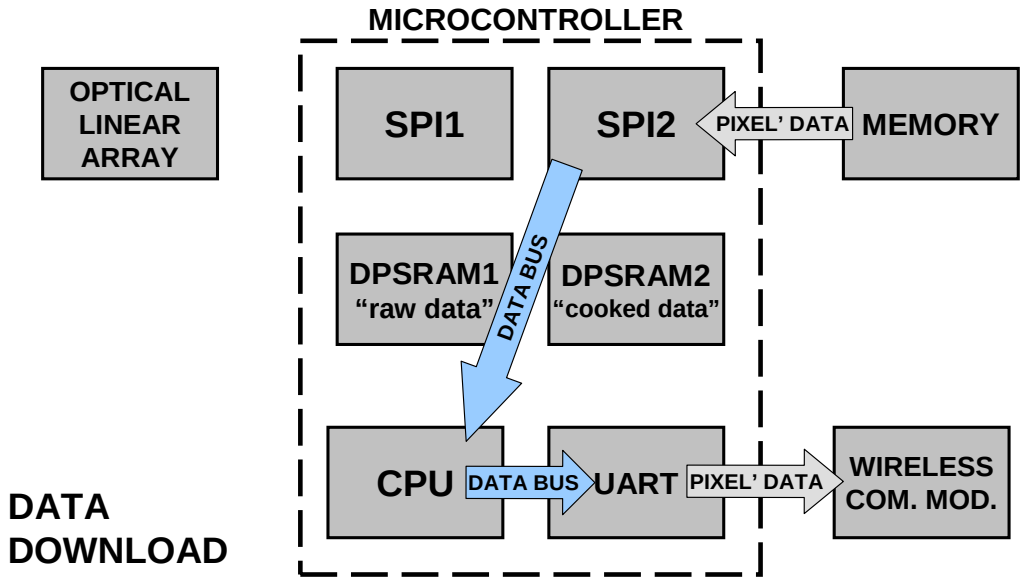

Fig. 22 Schematic diagram of hardware platform operation during data download

6.7 Demonstration of the Optical Linear Array Hardware Platform

In the spirit of “seeing is believing”, a demonstration of the operation of the optical linear array hardware platform is now presented. Recall that a linear optic array sees the world only along a single, 1-D line. However, if the linear array is scanned across a scene in a direction perpendicular to the longitudinal axis of the array, then a collection of 1-D images can be recorded, which if properly arranged can reconstruct the 2-deminisional (2-D) scene. Conversely, the linear array can remain stationary while the scene is scanned across the array’s field of view. This later scenario was exploited for this demonstration.

The apparatus for this demonstration is illustrated in Fig. 23. An image on a piece of paper was wrapped around a cylindrical mandrel that was chucked into an electric drill. The optical linear array, with focusing optics, was oriented to view the rotating image while data was acquired and logged. Following the approximately 0.1-s data acquisition event, the data was downloaded to a PC for long-term storage and visualization. The results of this operation are displayed in Fig. 24.

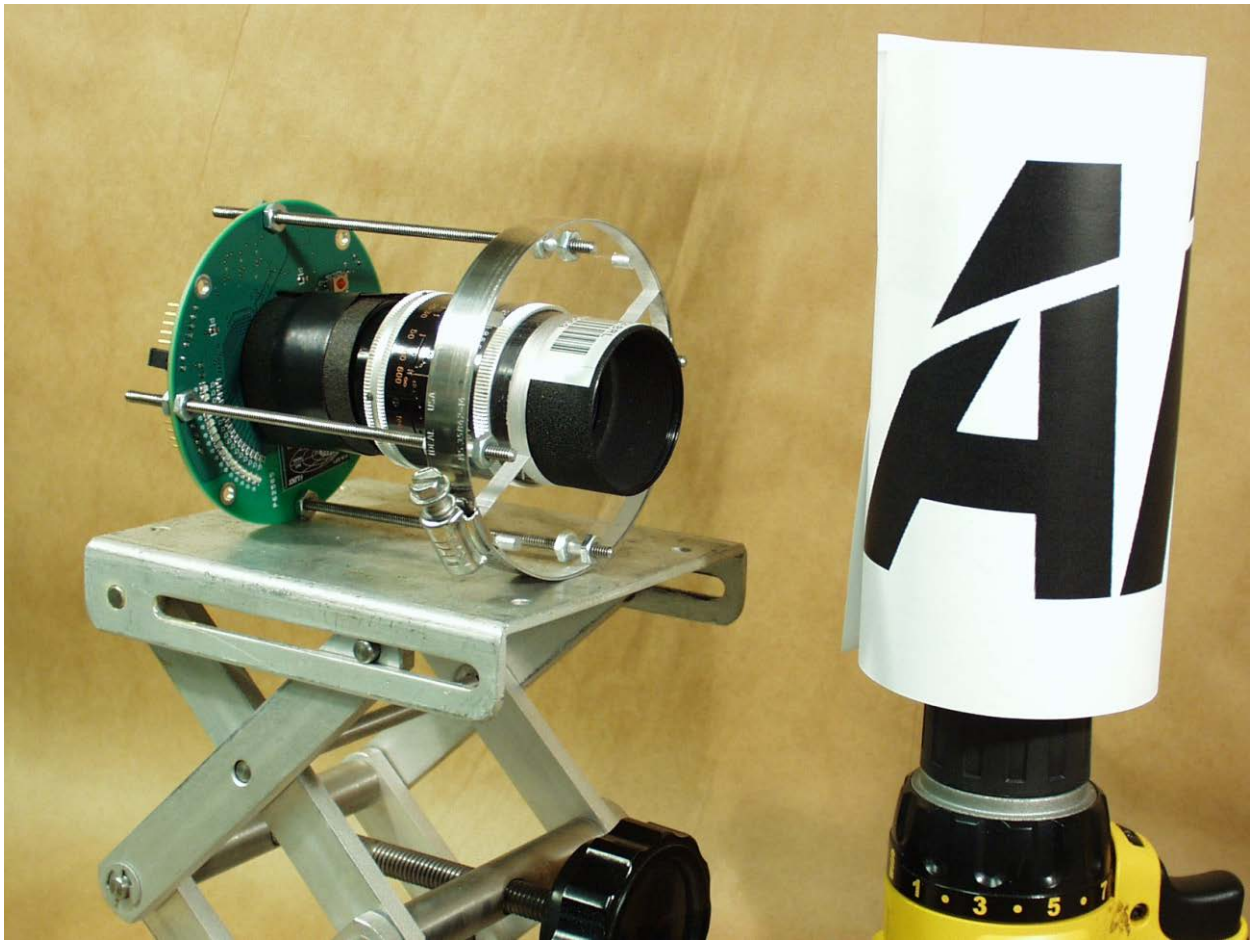

Fig. 23 Apparatus used for demonstration of optical linear array hardware platform

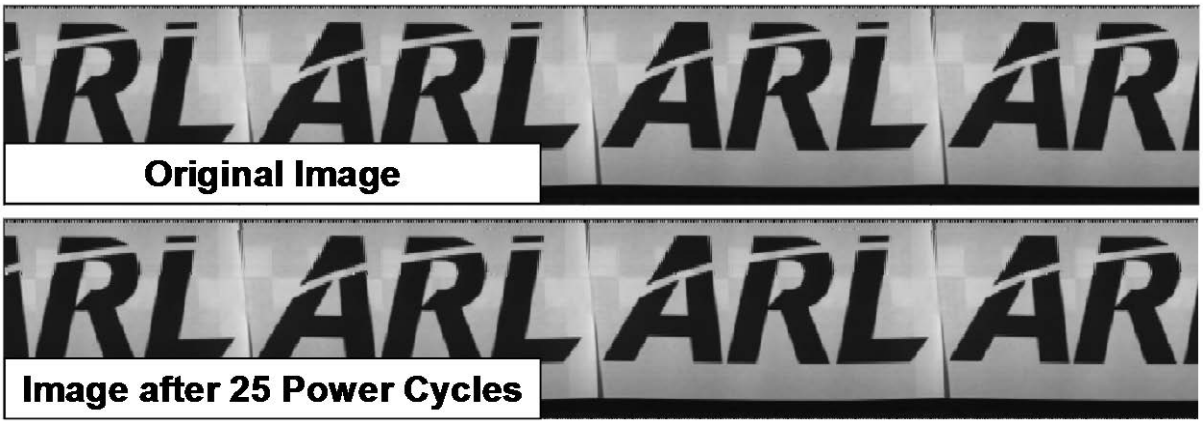

Fig. 24 Results of the demonstration of the optical linear array hardware platform

The upper image in Fig. 24 shows the reconstructed scene as originally recorded. During the approximately 0.1-s data acquisition event, the electric drill can be seen to rotate almost 4 times. To test the nonvolatility of the memory device, the power to the hardware platform was cycled on and off 25 times, and the logged data were retransmitted from the hardware platform to the PC. The results are displayed in the lower image of Fig. 24. The similarity between the upper and lower images of Fig. 24 indicates the data logging memory device is indeed nonvolatile. A closer look at Fig. 24 reveals a faint checkerboard pattern that is not obvious in Fig. 23. This pattern is the result of a checkerboard pattern that was on the mandrel to which the paper with the image was affixed. The ability of the optical linear array to detect this checkerboard pattern under the outer sheet of paper indicates that this linear array is capable of detecting subtle differences in shades of gray.

7. Summary

An optical hardware platform has been presented that is applicable to sensor systems utilizing a 1-D linear array. A basic goal in creating this apparatus was to provide a system that could simultaneously acquire, record, and analyze optical linear array data. The details of, and interactions between, the major hardware components, including the optical linear array, memory device, and microcontroller, have been presented in detail along with ancillary subsystems that support and enhance the functionality of this apparatus. Specifics concerning the design and fabrication of the hardware prototype have been highlighted. The operation of the optical linear array hardware platform has been explained in detail, including data acquisition, reformatting, analysis, monitoring, display, short-term onboard storage, and off-platform downloading. Finally, a simple demonstration of the apparatus' functionality has been presented.

8. References and Notes

1. Link to TLS3301 information page with data sheet. [accessed 2014 Aug]. http://www.ams.com /eng/Products.
2. Universal synchronous/asynchronous communications. Wikipedia. [accessed 2014 Aug]. http://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter.
3. Nonvolatile static random access memory. Wikipedia. [accessed 2014 Aug]. http://en.wikipedia.org/wiki/NVSRAM.
4. Serial port interface (SPI). Wikipedia. [accessed 2014 Aug]. http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus.
5. Link to CY14B101P information page with data sheet. [accessed 2014 Aug]. http://www.cypress.com/?rID=34815.
6. Link to PIC24HJ256GP210A data sheet. [accessed 2014 Aug]. http://www.microchip.com/www.products/Devices.aspx?dDocName=en541872.
7. Link to MPLABX information page with additional links to free download. [accessed 2014 Aug]. http://www.microchip.com/pagehandler/en-us/family/mplabx/home.html.
8. Link to MPLABXC information page with additional links to free download. [accessed 2014 Aug]. http://www.microchip.com/pagehandler/en_us/devtools/mplabxc/.
9. Bit-Banging Techniques. Wikipedia. [accessed 2014 Aug]. http://en.wikipedia.org/wiki/Bit-banging.
10. Link to ECS information page with additional link to data sheet. [accessed 2014 Aug]. http://www.ecsxtal.com/products/oscillators?id=207.
11. Phase Locked Loop. Wikipedia. [accessed 2014 Aug]. http://en.wikipedia.org/wiki/Wiki_/Phase-locked_loop.
12. Microchip Technology Inc. PIC24H family reference manual. Section 7 – Oscillator. Document number DS70227E, page 7–20. [accessed 2014 Aug]. Available at http://www.microchip.com /.
13. Three-state logic. Wikipedia. [accessed 2014 Aug] http://en.wikipedia.org/wiki/Three-state _logic.

14. Microchip Technology Inc. PIC24H family reference manual. Section 24.Programming and Diagnostics. Document number DS70246A, page 24-2. [accessed 2014 Aug]. Available at http://www.microchip.com /www.products/Devices.aspx?dDocName=en541872.

15. Debounce. Wiktionary. [accessed 2014 Aug]. http://en.wiktionary.org/wiki/debounce.

16. Link to Linx transceiver information page with additional links to data sheets. [accessed 2014 Aug]. https://www.linxtechnologies.com/en/products/modules/lt-rf-transceiver.

17. Link to Linx antenna information page with additional links to data sheets. [accessed 2014 Aug]. https://www.linxtechnologies.com/en/products/antennas/sp.

18. Link to MAX3232 data sheet MAX3222-MAX3241.pdf. [accessed 2014 Aug]. http://www.maximintegrated.com/en/products/interface/transceivers/MAX3232.html.

19. Microchip Technology Inc. PIC24H Family Reference Manual. Section 22 - Direct Memory Access (DMA). Document number DS70223B. [accessed 2014 Aug]. Available at http://www.microchip.com/wwwproducts/Devices.aspx?product=PIC24HJ256GP210A.

20. The concept of Endianness. Wikipedia. [accessed 2014 Aug] http://en.wikipedia.org/wiki/Wiki_/Endianness.

21. Microchip Technology Inc. PIC24H Family Reference Manual. Section 2 – CPU. Document number DS70245B, page 2-2. [accessed Aug 2014]. Available at http://www.microchip.com/wwwproducts /Devices .aspx?product=PIC24HJ256GP210A.

List of Symbols, Abbreviations, and Acronyms

APB	Applied Physics Branch
BPS	Bits per second
CPU	Central Processor Unit
CS	Chip select
DIG OUT	Digital annunciator output line
DMA	Direct memory access
DPSRAM	Dual-ported SRAM
FET	Field effect transistors
GPIO	General purpose input/output
Hold_	suspend serial operation
HSB_	hardware store pin
IDE	Integrated development environment
ICSP	In-circuit serial program
LD	Lethality Division
LED	Light emitting diode
LMB	Lethality Mechanisms Branch
LSByte	Less significant byte
Mbps	Megabits per second
MSByte	Most significant byte
NVSRAM	Nonvolatile static random access memory
PC	Personal computer
PCB	Printed circuit board
PD	Protection Division
PPL	Phase-locked loop

RAM	Random access memory
RC	Resistor-capacitor
SCK	Serial clock
SCLK	Serial clock line
SDIN	Serial data in
SDOUT	Serial data out
SI	Serial input
SO	Serial output
SRAM	Static random access memory
SPI	Serial port interface
UART	Universal asynchronous receiver transmitter
USART	Universal synchronous and asynchronous receiver transmitter
V	Volt
V_{cap}	Autostore capacitor
V_{cc}	Power supply , 2.7–3.6V
V_{ss}	Ground
WP_	Write protect memory
WMRD	Weapons and Material Research Directorate

Appendix. Schematic Diagrams for the Optical Linear Array Hardware Platform

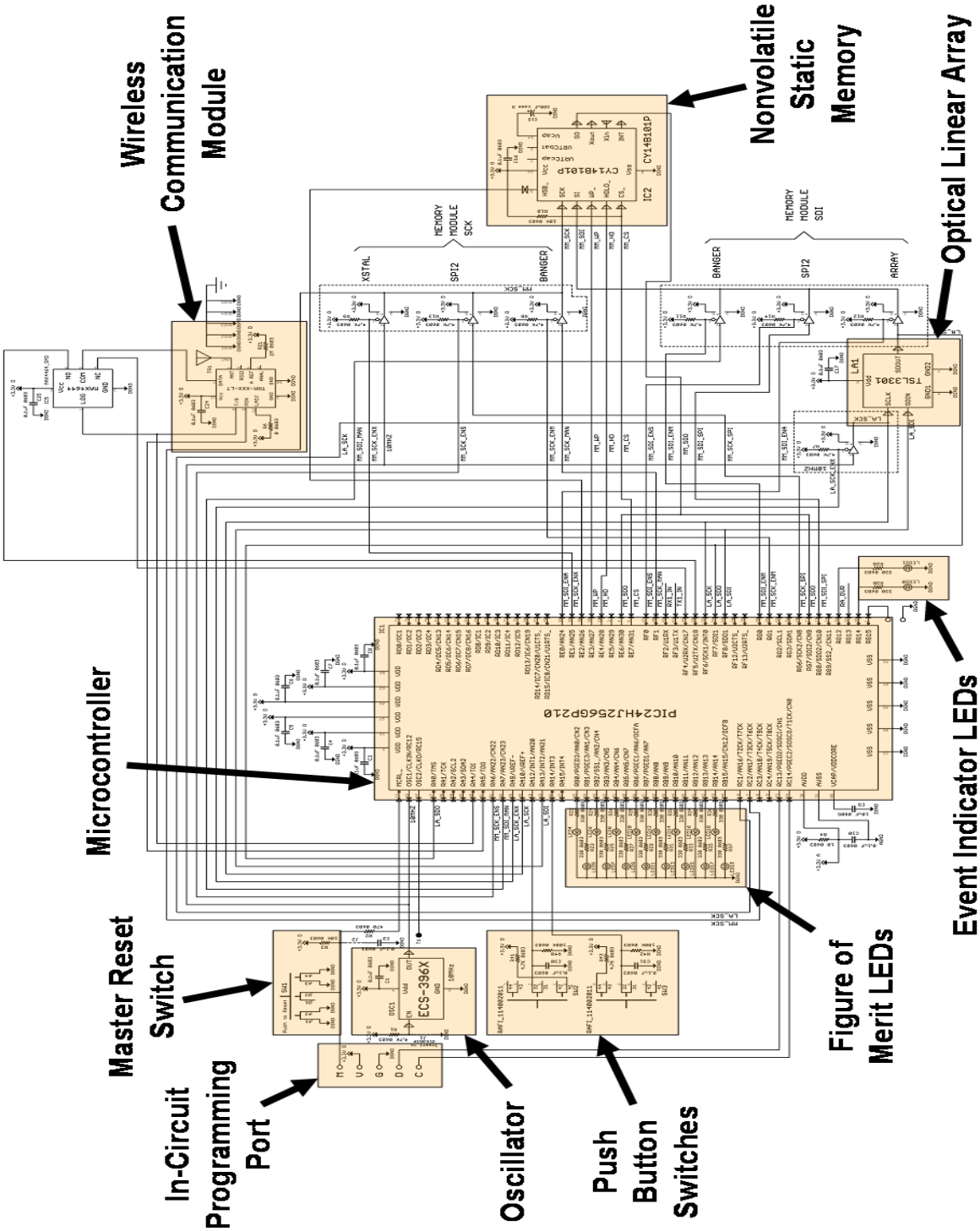

Fig. A-1 Schematic diagram of major and ancillary hardware components and their interconnections

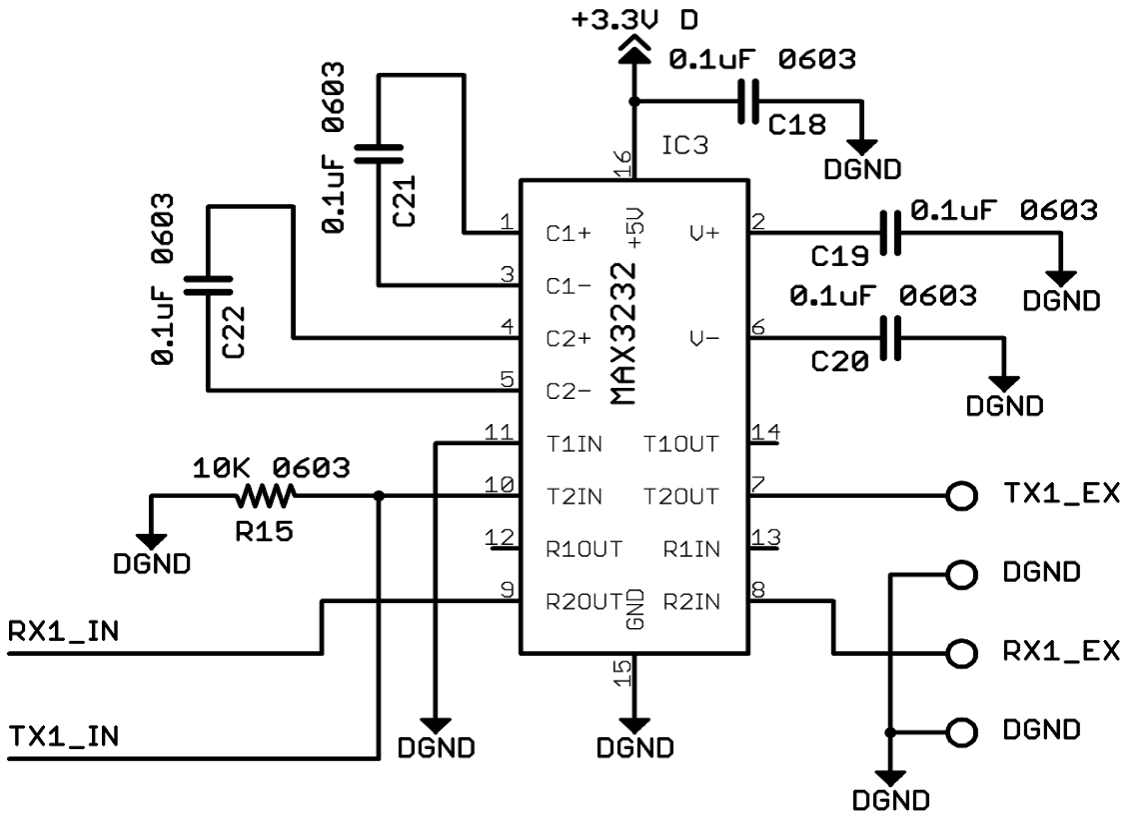

Fig. A-2 Schematic diagram of hard-wired communication module

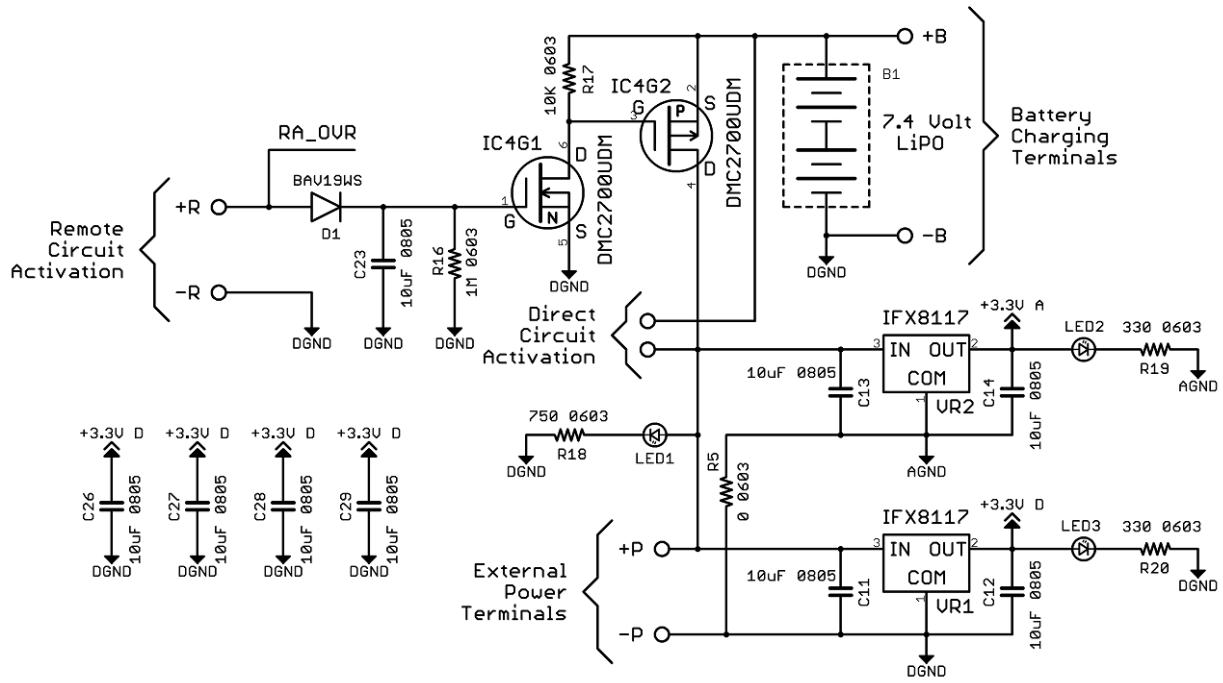

Fig. A-3 Schematic diagram of power management system

1 (PDF) DEFENSE TECHNICAL INFORMATION CTR DTIC OCA

2 (PDF) DIRECTOR US ARMY RESEARCH LAB RDRL CIO LL IMAL HRA MAIL & RECORDS MGMT

1 (PDF) GOVT PRINTG OFC A MALHOTRA

1 (PDF) HSI HYPERION SECURITY INC D MILLER

1 (PDF) SULLIVAN PRODUCTS J HUDSON

62 (56 PDF, 6 HC) DIR USARL
RDRL WM
P BAKER
R EHLERS
RDRL WML
P PEREGINO
M ZOLTOSKI
RDRL WML A
W OBERLE
RDRL WML B
N TRIVEDI
RDRL WML C
S AUBERT
RDRL WML D
D BEYER
RDRL WML E
P WEINACHT
RDRL WML F
G BROWN
R HALL
M HAMAOUI
M ILG
P MULLER
B TOPPER
RDRL WML G
J SOUTH
RDRL WML H
J ANGEL
M FERMEN-COKER
J NEWILL
RDRL WMM
R DOWDING
J ZABINSKI
RDRL WMM A
J SANDS
RDRL WMM B
P BARNES
RDRL WMM C
J LA SCALA
RDRL WMM D
R CARTER
RDRL WMM E
J P SINGH
RDRL WMM F
H MAUPIN
RDRL WMM G
A RAWLETT
J SNYDER
RDRL WMP
D LYON
S SCHOENFELD
T VONG
B YEAGER
RDRL WMP A
J BALL
P BERNING
M COPPINGER
J FLENIKEN
C HUMMER
T KOTTKE (6 HC)
M MCNEIR
A PORWITZKY
G THOMSON
W UHLIG
A VALENZUELA
C WOLFE
RDRL WMP B
C HOPPEL
RDRL WMP C
T BJERKE
R LEAVY
R MUDD
RDRL WMP D
J RUNYEON
M ZELLNER
RDRL WMP E
P BARTKOWSKI
D HACKBARTH
P SWOBODA
RDRL WMP F
N GNIAZDOWSKI
RDRL WMP G
N ELDREDGE